

PROYEK AKHIR - VS191801

**KLASIFIKASI SENTIMEN MASYARAKAT TERHADAP
KEPOLISIAN NEGARA REPUBLIK INDONESIA
MENGUNAKAN *NAÏVE BAYES CLASSIFIER* DAN
*SUPPORT VECTOR MACHINE***

NADA DAMAI BAGASKARA
NRP 2043211117

Dosen Pembimbing
Mike Prastuti, S.Si., M.Si.
NIP 19910122 201504 2 002

Program Studi Sarjana Terapan
Departemen Statistika Bisnis
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya
2022



PROYEK AKHIR - VS191801

**KLASIFIKASI SENTIMEN MASYARAKAT TERHADAP
KEPOLISIAN NEGARA REPUBLIK INDONESIA
MENGUNAKAN *NAÏVE BAYES CLASSIFIER* DAN
*SUPPORT VECTOR MACHINE***

NADA DAMAI BAGASKARA
NRP 2043211117

Dosen Pembimbing
Mike Prastuti, S.Si., M.Si.
NIP 19910122 201504 2 002

Program Studi Sarjana Terapan
Departemen Statistika Bisnis
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya
2022



FINAL PROJECT - VS191801

**SOCIETY SENTIMENT CLASSIFICATION OF INDONESIAN
POLICE INSTITUTION USING NAÏVE BAYES CLASSIFIER
AND SUPPORT VECTOR MACHINE**

NADA DAMAI BAGASKARA
NRP 2043211117

Advisor
Mike Prastuti, S.Si., M.Si.
NIP 19910122 201504 2 002

Applied Undergraduate Study Programme
Business Statistics Department
Faculty of Vocational
Institut Teknologi Sepuluh Nopember
Surabaya
2022

LEMBAR PENGESAHAN

KLASIFIKASI SENTIMEN MASYARAKAT TERHADAP KEPOLISIAN NEGARA
REPUBLIK INDONESIA MENGGUNAKAN *Naïve Bayes Classifier* DAN
SUPPORT VECTOR MACHINE

TUGAS AKHIR

Diajukan untuk memenuhi salah satu syarat
Memperoleh gelar Sarjana Terapan pada
Program Studi Sarjana Terapan
Departemen Statistika Bisnis
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember

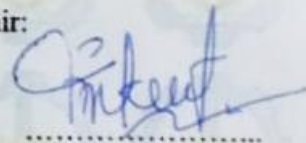
Oleh: **NADA DAMAI BAGASKARA**
NRP. 2043211117

Disetujui oleh Tim Penguji Tugas Akhir:

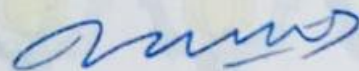
Mike Prastuti, S.Si., M.Si.
Pembimbing

Dr. Wahyu Wibowo, S.Si., M.Si.
Penguji

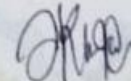
Zakiatul Wildani, S.Si., M.Sc
Penguji



.....



.....



.....

SURABAYA
Juli, 2022

APPROVAL SHEET

**SOCIETY SENTIMENT CLASSIFICATION OF INDONESIAN POLICE
INSTITUTION USING NAÏVE BAYES CLASSIFIER AND SUPPORT VECTOR
MACHINE**

FINAL PROJECT

Submitted to fulfill one of the requirements
For obtaining an Applied Undergraduate Degree at
Study Programme of Applied Undergraduate
Department of Business Statistics
Faculty of Vocational
Institut Teknologi Sepuluh Nopember

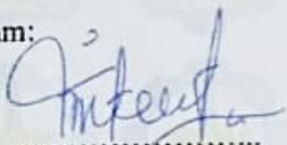
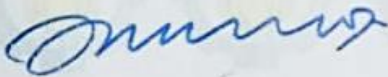
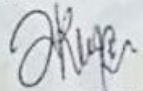
By: **NADA DAMAI BAGASKARA**
NRP. 2043211117

Approved by Final Project Examiner Team:

Mike Prastuti, S.Si., M.Si.
Advisor

Dr. Wahyu Wibowo, S.Si., M.Si.
Examiner

Zakiatul Wildani, S.Si., M.Sc
Examiner


.....

.....

.....

SURABAYA
July, 2022

PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini;

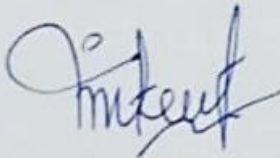
Nama mahasiswa/NRP : Nada Damai Bagaskara / 2043211117
Departemen : Statistika Bisnis
Dosen Pembimbing/NIP : Mike Prastuti, S.Si., M.Si. / 19910122 201504 2 002

dengan ini menyatakan bahwa Tugas Akhir dengan judul "Klasifikasi Sentimen Masyarakat terhadap Kepolisian Negara Republik Indonesia menggunakan *Naïve Bayes Classifier* dan *Support Vector Machine*" adalah hasil karya sendiri, bersifat orisinal, dan ditulis dengan mengikuti kaidah penulisan ilmiah.

Bilamana di kemudian hari ditemukan ketidaksesuaian dengan pernyataan ini, maka saya bersedia menerima sanksi sesuai dengan ketentuan yang berlaku di Institut Teknologi Sepuluh Nopember.

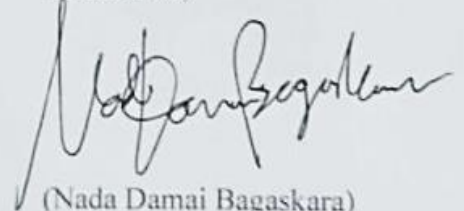
Surabaya, 1 Juli 2022

Mengetahui
Dosen Pembimbing



(Mike Prastuti, S.Si., M.Si.)
NIP. 19910122 201504 2 002

Mahasiswa,



(Nada Damai Bagaskara)
NRP. 2043211117

STATEMENT OF ORIGINALITY

The undersigned below:

Name of student /NRP : Nada Damai Bagaskara / 2043211117
Departement : Business Statistics
Advisor/NIP : Mike Prastuti, S.Si., M.Si. / 19910122 201504 2 002

Hereby declare that the Final Project with the title of "Society Sentiment Classification of Indonesian Police Institution using Naïve Bayes Classifier and Support Vector Machine" is the result of my own work, is original, and is written by following the rules of scientific writing. If in the future there is a discrepancy with this statement, then I am willing to accept sanctions in accordance with the provisions that apply at Institut Teknologi Sepuluh Nopember.

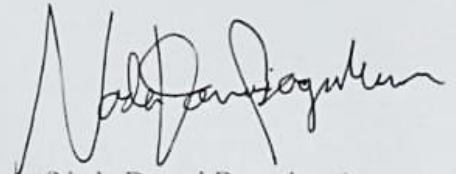
Surabaya, 1st July 2022

Supervised by
Advisor



(Mike Prastuti, S.Si., M.Si.)
NIP. 19910122 201504 2 002

Student,



(Nada Damai Bagaskara)
NRP. 2043211117

KLASIFIKASI SENTIMEN MASYARAKAT TERHADAP KEPOLISIAN NEGARA REPUBLIK INDONESIA MENGGUNAKAN *NAÏVE BAYES CLASSIFIER* DAN *SUPPORT VECTOR MACHINE*

Nama mahasiswa/NRP : Nada Damai Bagaskara / 2043211117

Departemen : Statistika Bisnis ITS

Dosen Pembimbing : Mike Prastuti, S.Si., M.Si.

Abstrak

Pada tahun 2021 sejumlah anggota kepolisian dinilai beberapa kali melakukan tindakan diluar kode etik dan norma kemanusiaan. Puncaknya adalah akhir tahun 2021 dengan munculnya tagar #percumalaporpolisi, kemunculan tagar ini didorong karena peristiwa tindakan asusila dari beberapa orang anggota polisi. Berdasarkan kejadian-kejadian tersebut, perlu diketahui sentimen masyarakat terhadap Kepolisian Negara Republik Indonesia. Hal tersebut dilakukan karena lembaga Kepolisian Negara Republik Indonesia haruslah memiliki wibawa dan kepercayaan dimata masyarakat, sehingga perlu diketahui ada tidaknya kepercayaan masyarakat terhadap lembaga tersebut. Kepolisian Negara Republik Indonesia sebagai sebuah lembaga perlu menjaga kepercayaan masyarakat, dimana disisi lain masyarakat juga perlu menyampaikan kritik agar lembaga kepolisian tersebut berbenah. Pada penelitian ini dilakukan analisis sentimen dimana data diambil dengan *crawling* Twitter, menggunakan kata kunci polisi dan polri untuk mendapatkan ulasan masyarakat mengenai kepolisian. Klasifikasi yang digunakan adalah *Naïve Bayes Classifier* (NBC) dan *Support Vector Machine* (SVM), dimana selanjutnya dibandingkan untuk mendapatkan metode klasifikasi terbaik. Hasil dari penelitian ini menunjukkan bahwa sentimen negatif jauh lebih banyak muncul dibandingkan positif, dengan persentase 80% banding 20%, dimana NBC lebih unggul daripada SVM dengan skor AUC sebesar 0,6410 dan 0,6361 pada ketepatan klasifikasi data *training* dan data *testing*.

Kata Kunci : Analisis Sentimen, *Naïve Bayes Classifier*, *Support Vector Machine*

SOCIETY SENTIMENT CLASSIFICATION OF INDONESIAN POLICE INSTITUTION USING NAÏVE BAYES CLASSIFIER AND SUPPORT VECTOR MACHINE

Student Name/NRP : Nada Damai Bagaskara / 2043211117
Departement : Business Statistics ITS
Advisor : Mike Prastuti, S.Si., M.Si.

Abstrak

In the year of 2021 a number of police officers were assessed several times for taking actions outside the code of ethics and humanitarian norms. The climax is at the end of 2021 with the emergence of the hashtag #percumalaporpolisi, the emergence of this hashtag was driven by the immoral acts of several police officers. Based on these incidents, it is necessary to know the public sentiment towards the of the Indonesian National Police. This is done because the National Police of the Republic of Indonesia must have authority and trust in the eyes of the public, so it is necessary to know whether there is public trust in the institution. The National Police of the Republic of Indonesia as an institution needs to maintain public trust, where on the other hand the public also needs to express criticism so that the police institution can improve. In this study, a sentiment analysis was carried out where the data was taken by crawling Twitter, using the keywords police and police to get public reviews about the of the police. The classifications used are Naïve Bayes Classifier (NBC) and Support Vector Machine (SVM), which are then compared to obtain the best classification method. The results of this study indicate that negative sentiments are much more prevalent than positive, with a percentage of 80% versus 20%, where NBC is superior to SVM with AUC scores of 0.6410 and 0.6361 on the accuracy of training data classification and testing data.

Keywords: *Naïve Bayes Classifier, Sentiment Analysis, Support Vector Machine*

KATA PENGANTAR

Kemuliaan bagi Allah Sang Bapa, Sang Putra, dan Sang Roh Kudus sekarang, selalu, serta sepanjang segala abad. Karunia Tuhan Allah yang telah memberikan rahmat dan kebijaksanaan kepada penulis untuk menyelesaikan Proyek Akhir dengan judul “Klasifikasi Sentimen Masyarakat terhadap Kepolisian Negara Republik Indonesia menggunakan *Naïve Bayes Classifier* dan *Support Vector Machine*”. Proyek Akhir ini ditulis dengan proses yang tidak mudah, dan menghadapi banyak tantangan. Banyak pihak yang turut berkontribusi dalam terciptanya penelitian Proyek Akhir ini, sehingga penulis ingin menyampaikan banyak-banyak terima kasih kepada:

1. Dr. Wahyu Wibowo, S.Si., M.Si, sebagai Kepala Departemen Statistika Bisnis Fakultas Vokasi Institut Teknologi Sepuluh Nopember (ITS), sekaligus dosen penguji Proyek Akhir.
2. Mike Prastuti, S.Si., M.Si, sebagai dosen pembimbing Proyek Akhir.
3. Zakiatul Wildani, S.Si., M.Sc. sebagai dosen penguji Proyek Akhir.
4. Dra. Lucia Aridinanti, M.T. sebagai dosen wali.
5. Kedua orang tua penulis beserta seluruh keluarga penulis.
6. Seluruh civitas akademika Departemen Statistika Bisnis ITS.
7. Angkatan 2016 Berdikari.
8. Angkatan LJ 2021.

Kiranya penelitian ini dapat memberi manfaat bagi dunia akademis dan bagi pembaca umum, khususnya di era jaringan internet global saat ini. Mudah-mudahan metode ilmiah berbasis data dapat dipahami dan digunakan masyarakat secara luas

Surabaya, 1 Juli 2022

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	vii
ABSTRAK	xv
KATA PENGANTAR	xix
DAFTAR ISI	xxi
DAFTAR GAMBAR	xxiii
DAFTAR TABEL	xxv
DAFTAR LAMPIRAN	xxvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian.....	2
BAB II TINJAUAN PUSTAKA	3
2.1 Penelitian Sebelumnya	3
2.2 Landasan Teori	3
2.2.1 <i>Text Mining</i>	3
2.2.2 Pra Proses Teks	4
2.2.3 Analisis Sentimen	4
2.2.4 <i>Term Frequency Inverse Document Frequency (TF-IDF)</i>	4
2.2.5 <i>Naïve Bayes Classifier (NBC)</i>	5
2.2.6 <i>Support Vector Machine (SVM)</i>	6
2.2.7 <i>Holdout Validation</i>	9
2.2.8 Ketepatan Klasifikasi	9
2.2.9 <i>Word Cloud</i>	10
2.3 Polisi.....	10
2.4 Twitter	11
2.5 Citra Kepolisian.....	11
BAB III METODOLOGI PENELITIAN	13
3.1 Metode Pengumpulan Data	13
3.2 Struktur Data dan Variabel Penelitian.....	13
3.3 Metode Penelitian.....	13
3.4 Langkah Analisis	14
3.5 Diagram Alir.....	15
BAB IV ANALISIS DAN PEMBAHASAN	17
4.1 Pra Proses Data <i>Tweet</i>	17
4.1.1 Pemberian Skor pada Data <i>Tweet</i>	17

4.1.2	Pembersihan Data <i>Tweet</i>	18
4.2	Karakteristik Data Sentimen Masyarakat Terhadap Kepolisian Negara Republik Indonesia	19
4.3	Klasifikasi menggunakan <i>Naïve Bayes Classifier</i>	21
4.4	Klasifikasi menggunakan <i>Support Vector Machine</i>	23
4.5	Pemilihan Metode Klasifikasi Terbaik	24
BAB V KESIMPULAN DAN SARAN		27
5.1	Kesimpulan	27
5.2	Saran	27
DAFTAR PUSTAKA		29
LAMPIRAN.....		31
BIODATA PENULIS.....		55

DAFTAR GAMBAR

Gambar 2. 1 Alternatif Bidang Pemisah (Kiri) dan Bidang Pemisah Terbaik dengan Margin (m) Terbesar (Kanan).....	6
Gambar 2. 2 Ruang Data Dua Dimensi dengan Pemisah Non Linier (Kiri) dan Ruang Fitur Tiga Dimensi dengan Pemisah Linier (Kanan).....	8
Gambar 2. 3 Pembagian Data menggunakan Holdout Validation	9
Gambar 2. 4 Contoh Word Cloud sentimen positif (kiri) dan sentimen negatif (kanan).....	10
Gambar 3. 1 Diagram Alir	15
Gambar 4. 1 Jumlah Tweet per Kelas	19
Gambar 4. 2 Frekuensi Kata Teratas.....	20
Gambar 4. 3 Wordcloud Data Tweet Keseluruhan	20
Gambar 4. 4 Wordcloud Tweet Skor Positif.....	21
Gambar 4. 5 Wordcloud Tweet Skor Negatif	21
Gambar 4. 6 Hasil Parameter C dan Gamma Terbaik.....	23

DAFTAR TABEL

Tabel 2. 1 Confussion Matrix	9
Tabel 2. 2 Kategori Skor AUC	10
Tabel 3. 1 Contoh Struktur Data	13
Tabel 3. 2 Variabel Penelitian.....	13
Tabel 4. 1 Pemberian Skor.....	17
Tabel 4. 2 Pembersihan Data Tweet	18
Tabel 4. 3 Data Bersih	18
Tabel 4. 4 Term Frequency (TF)	19
Tabel 4. 5 Ketepatan Klasifikasi NBC.....	22
Tabel 4. 6 Proporsi Tweet.....	22
Tabel 4. 7 TF-IDF.....	23
Tabel 4. 8 Ketepatan Klasifikasi SVM	24
Tabel 4. 9 Ketepatan Klasifikasi Data Training NBC dan SVM.....	24
Tabel 4. 10 Ketepatan Klasifikasi Data Testing NBC dan SVM.....	25

DAFTAR LAMPIRAN

Lampiran 1 Syntax R Pelabelan Kelas Sentimen	31
Lampiran 2 Syntax R Pra Proses Data.....	33
Lampiran 3 Syntax R TF	35
Lampiran 4 Syntax R TF-IDF.....	36
Lampiran 5 Syntax R Holdout Validation	37
Lampiran 6 Syntax R Naïve Bayes Classifier	39
Lampiran 7 Syntax Python Pencarian Parameter C dan Gamma Terbaik.....	41
Lampiran 8 Syntax R Support Vector Machine	43
Lampiran 9 Syntax R Word Cloud	44
Lampiran 10 Output R NBC.....	45
Lampiran 11 Output R SVM	50

BAB I PENDAHULUAN

1.1 Latar Belakang

Pelayanan masyarakat dalam hal pertahanan dan keamanan bernegara tentu menjadi tanggung jawab pemerintah dengan dinaungi oleh aparat penegak hukum. Pertahanan dan keamanan bertujuan agar seluruh aktivitas masyarakat mulai dari pusat, daerah, hingga hubungan internasional dapat berjalan dengan lancar tanpa gangguan. Masyarakat juga memiliki tempat mengadu dan berlindung ketika menghadapi masalah seperti kriminalitas, kehilangan barang berharga, orang hilang, bencana alam, dan lain sebagainya. Adanya aparat penegak hukum diharapkan mampu menjadi pengayom masyarakat dengan memberikan perlindungan, rasa aman, dan respon cepat tanggap terhadap masalah. Hal tersebut menunjukkan bahwa aparat penegak hukum memiliki tanggung jawab moral yang besar, sekaligus merupakan profesi yang mulia.

Salah satu aparat penegak hukum di Indonesia adalah polisi, yang memiliki tanggung jawab tugas menjaga keamanan dan kondusifitas masyarakat khususnya tingkat sipil. Pada tingkat nasional, polisi-polisi di Indonesia berada dibawah naungan lembaga otonom pemerintah yakni Kepolisian Negara Republik Indonesia (POLRI). Secara umum beberapa contoh tugas polisi adalah menyelidiki kasus kriminalitas, menangkap penjahat, mencari orang hilang, mengatur lalu lintas kendaraan, serta melindungi keamanan penduduk sipil.

Pada akhir tahun 2020, terjadi gejolak politik di Indonesia karena disebabkan oleh berbagai macam kondisi. Penyebab-penyebab gejolak tersebut antara lain karena adanya pandemi *Corona Virus Disease 19* (Covid-19), masuknya tenaga kerja asing, Undang-Undang Cipta Kerja, kasus korupsi bantuan dana sosial pandemi Covid-19 yang didalangi oleh mantan Menteri Sosial Republik Indonesia, adanya undang-undang yang oleh masyarakat dinilai melemahkan Komisi Pemberantasan Korupsi Republik Indonesia (KPK-RI), serta berbagai permasalahan lainnya. Masalah-masalah tersebut mendapat respon dari masyarakat baik kritik di media sosial, kritik lisan, maupun aksi turun ke jalan. Berbagai macam respon tersebut diberikan oleh masyarakat dari berbagai macam kalangan seperti profesional, akademisi, politisi, mahasiswa, hingga masyarakat sipil. Hal-hal tersebut tentu merupakan kondisi dimana aparat kepolisian juga turut andil dalam menjaga stabilitas bernegara (BBC, 2020).

Pada kondisi di lapangan terdapat banyak kritik dari masyarakat terhadap kepolisian khususnya pada tahun 2021. Beberapa aparat kepolisian dinilai tidak memberikan pengayoman yang baik, tidak menjalankan kode etik, dan sebagainya. Contoh kasus yang kerap dilakukan beberapa aparat kepolisian sehingga mendapat kritik masyarakat adalah anarkisme terhadap masyarakat sipil, dimana aksi pendisiplinan masyarakat dilakukan dengan kekerasan seperti menghajar juga memukul masyarakat secara fisik (CNN, 2021). Kasus lain yang menjadi kecaman adalah aksi pembantingan tubuh mahasiswa hingga kejang-kejang saat demonstrasi (BBC, 2021). Hingga pada akhir tahun 2021 terdapat kasus pemerkosaan yang dilakukan salah seorang anggota kepolisian (Kompas, 2022), dan kemudian disusul oleh kasus asusila lainnya. Keluhan-keluhan lain yang disampaikan masyarakat terkait penanganan pengaduan masyarakat dinilai lamban, dimana terdapat beberapa kasus hanya diminta keterangan lalu tidak ditindak lanjuti apa-apa (Brawijaya, 2022). Kasus-kasus tersebut tentu masih terekam jelas pada memori masyarakat, apalagi pada jejak digital.

Berdasarkan pemaparan contoh kasus yang dilakukan oleh beberapa anggota kepolisian di Indonesia, perlu diketahui persepsi masyarakat terhadap Kepolisian Negara Republik Indonesia. Hal tersebut dilakukan karena lembaga Kepolisian Negara Republik Indonesia haruslah memiliki wibawa dan kepercayaan dimata masyarakat, sehingga perlu diketahui ada tidaknya kepercayaan masyarakat terhadap lembaga tersebut. Permasalahan ini dapat dilakukan

dengan menganalisis data yang diambil dari Twitter dengan kata kunci polisi dan polri, dimana selanjutnya dilakukan *scrapping* untuk mengambil data. Metode yang digunakan adalah analisis sentimen dengan *Naïve Bayes Classifier* serta *Support Vector Machine*. *Naïve Bayes Classifier* merupakan algoritma yang berguna untuk mendapatkan klasifikasi teks dalam penelitian ini, dimana algoritma tersebut memiliki bahasa yang lebih sederhana. Algoritma *Naïve Bayes Classifier* memiliki asumsi bahwa tiap kata dalam data teks tidak saling terkait dengan kata lainnya atau bersifat independen. *Support Vector Machine* merupakan serangkaian metode dari pembelajaran yang terawasi (*supervised learning*). Metode tersebut berguna dalam analisis data dengan mengenali pola untuk mencari klasifikasi yang tepat. Terdapat penelitian sebelumnya sebagai acuan dan referensi yang juga mengambil data dari Twitter mengenai media TVOne, Metro Tv, dan Kompas TV (Kurniawan, 2017). Penelitian sebelumnya yang kedua adalah analisis sentimen masyarakat terhadap *E-Commerce* pada Media Sosial (Syakuro, 2017). Penelitian sebelumnya yang ketiga adalah analisis sentimen menggunakan *Support Vector Machine* (Nomleni, 2015). Ketiga penelitian tersebut melakukan analisis sentimen dari data Twitter serta klasifikasi menggunakan *Naïve Bayes Classifier* dan atau *Support Vector Machine*.

1.2 Rumusan Masalah

Permasalahan pada Kepolisian Negara Republik Indonesia adalah terjadinya beberapa tindakan represif kepada masyarakat, munculnya pelanggaran asusila, hingga pelayanan terhadap laporan masyarakat yang tidak ditanggapi oleh sebagian anggota kepolisian. Hal tersebut dianggap mempengaruhi citra lembaga kepolisian dimata masyarakat, oleh karena itu diperlukan sebuah analisis terhadap persepsi masyarakat terhadap lembaga Kepolisian Negara Republik Indonesia.

1.3 Batasan Masalah

Batasan masalah pada penelitian ini adalah sebagai berikut.

1. Menggunakan kata kunci polisi, dan polri sebagai kata kunci.
2. Sentimen yang digunakan dalam penelitian ini adalah kelas positif dan kelas negatif, dimana kelas netral tidak digunakan karena tidak membantu dalam pengambilan keputusan.

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah tersebut diperoleh tujuan pada penelitian ini adalah sebagai berikut.

1. Mendapatkan hasil persepsi masyarakat terhadap lembaga Kepolisian Negara Republik Indonesia.
2. Mendapatkan hasil ketepatan klasifikasi terbaik dengan membandingkan antara *Naïve Bayes Classifier* dan *Support Vector Machine*.

1.5 Manfaat Penelitian

Hasil penelitian ini diharapkan dapat membantu lembaga Kepolisian Republik Indonesia untuk lebih memahami persepsi masyarakat terhadap rekam jejak kepolisian. Manfaat bagi masyarakat ialah sebagai sarana mengutarakan tanggapan secara ilmiah kepada lembaga Kepolisian Negara Republik Indonesia, sehingga diharapkan dapat memperbaiki kinerja, perilaku, dan kode etik.

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Sebelumnya

Penelitian sebelumnya digunakan pada penelitian ini sebagai acuan, referensi, serta teori pendukung. Judul penelitian pertama yang digunakan sebagai acuan adalah “Implementasi *Text Mining* pada Analisis Sentimen Pengguna Twitter terhadap Media *Mainstream* menggunakan *Naïve Bayes Classifier* dan *Support Vector Machine*”. Hasil dari penelitian tersebut menunjukkan bahwa metode *Support Vector Machine* lebih baik dalam klasifikasi data daripada *Naïve Bayes Classifier* secara keseluruhan. Ketepatan klasifikasi dengan *Naïve Bayes Classifier* untuk media TV One dan Kompas TV menunjukkan akurasi 95,8% dan 97,8%, sementara media Metro TV mendapatkan nilai *G-mean* sebesar 81,3% dan AUC sebesar 82,36%. Ketepatan klasifikasi dengan *Support Vector Machine* untuk media TV One dan Kompas TV menunjukkan akurasi 97,9% dan 99,3%, sementara media Metro TV mendapatkan nilai *G-mean* sebesar 97,35% dan AUC sebesar 97,38%. *Support Vector Machine* menggunakan kernel RBF menghasilkan ketepatan klasifikasi yang lebih baik daripada kernel linier (Kurniawan, 2017).

Judul penelitian kedua adalah “Analisis Sentimen Masyarakat terhadap *E-Commerce* pada Media Sosial menggunakan Metode *Naïve Bayes Classifier (NBC)* dengan Seleksi Fitur *Information Gain (IG)*”. Hasil dari penelitian tersebut menunjukkan bahwa penggunaan *Naïve Bayes Classifier* dengan Seleksi Fitur *Information Gain* terbukti secara otomatis melakukan analisis sentimen. Kombinasi dua metode tersebut mengalami performa yang meningkat pada batas *threshold* 2200 waktu proses menjadi 514 detik lebih cepat dibandingkan tanpa *Information Gain*, serta menambah akurasi sebesar 88,8% pada sistem analisis sentimen (Syakuro, 2017).

Judul penelitian ketiga adalah “*Sentiment Analysis* menggunakan *Support Vector Machine (SVM)*”. Hasil dari penelitian tersebut menunjukkan bahwa dari tujuh kali percobaan diperoleh rata-rata akurasi diatas 80%, dengan akurasi tertinggi sebesar 84,4086%, menggunakan perbandingan jumlah data *training* dan data *testing* sebesar 20% dan 80%. Penentuan nilai data setelah *stemming* berpengaruh terhadap hasil klasifikasi, karena akurasi lebih rendah ketika data sudah *stemming* (Nomleni, 2015).

2.2 Landasan Teori

Sub bab ini menjelaskan teori-teori yang digunakan dalam penelitian ini, baik secara formula maupun definisi. Penjelasan teori-teori diambil berdasarkan pengertian dari peneliti-peneliti sebelumnya.

2.2.1 *Text Mining*

Text Mining merupakan proses menggali atau menambang data berupa teks yang diperoleh dari sebuah dokumen untuk mendapatkan informasi dari kata-kata yang merepresentasikan isi dokumen tersebut serta hubungan yang terbentuk. Kebutuhan informasi dapat terbantu oleh proses ini, yaitu dengan menerapkan teknik *data mining*, *machine learning*, *natural language processing*, mencari informasi, serta manajemen pengetahuan (Feldman & Sanger, 2007). Metode tersebut dapat digunakan untuk penanganan klasifikasi, pembentukan kluster, ekstraksi dan pengambilan informasi (Berry & Kogan, 2010) Data teks tersebut dapat berupa, kata, frasa, kalimat, hingga narasi panjang, oleh sebab itu teks dikategorikan sebagai data tidak terstruktur sehingga membutuhkan tahap pra proses teks. Proses tersebut dapat digunakan antara lain untuk mengetahui sentimen yang terkandung dalam teks, mendeteksi pesan spam, mengorganisasi dokumen, dan lain-lain.

2.2.2 Pra Proses Teks

Pra proses teks merupakan serangkaian tahapan untuk mengolah data teks yang pada awalnya tidak terstruktur menjadi terstruktur sehingga siap digunakan atau dianalisis. Beberapa contoh tahapan dalam pra proses teks ialah menghapus kata atau karakter yang dianggap tidak penting, seperti halnya salah pengetikan, pengetikan ganda, singkatan, kata hubung, dan lain-lain. Bentuk data teks yang terstruktur serta siap dianalisis biasanya telah tersaji dalam tabel yang memiliki baris dan kolom (Weiss, 2010). Tahapan-tahapan dalam melakukan pra proses teks adalah sebagai berikut.

a. *Case Folding*

Case folding merupakan tahapan dalam mengubah semua kata-kata menjadi huruf kecil, serta menghapus karakter non huruf seperti angka dan tanda baca. Algoritma ini hanya memproses karakter huruf “a” sampai “z”, sehingga selain karakter tersebut akan dihilangkan (Weiss, 2010).

b. Tokenisasi

Tokenisasi merupakan proses menjadikan teks yang utuh menjadi kata-kata terpisah, dengan cara memutuskan urutan string sehingga diperoleh potongan-potongan kata dari teks tersebut.

c. Menghilangkan *Stop Words*

Stop Words adalah sekumpulan kata-kata yang dianggap tidak bermakna unik atau tidak menyampaikan pesan signifikan, sehingga keberadaannya dianggap tidak memberikan pengaruh apapun terhadap arti kalimat. Hal tersebut membuat *stop words* perlu dihapus sehingga tidak terjadi penumpukan kata tidak bermakna saat dilakukan analisis. Contoh *stop words* antara lain dan, atau, untuk, dari, akan, dan sebagainya.

d. *Stemming*

Stemming adalah proses untuk mendapatkan kata dasar, sehingga kata-kata yang berimbuhan akan dihilangkan imbuhan. Contoh proses *stemming* untuk kata “mempercai” dimana imbuhan depan “mem” dan imbuhan belakang “i” dihilangkan, sehingga diperoleh kata dasar “percai”.

e. Menghilangkan *Link* dan URL

Data teks yang diambil dari Twitter sangat rentang mengandung *link*, URL, tagar (#), nama akun dalam “@”, dan semacamnya, sehingga obyek-obyek tersebut perlu dihilangkan untuk mendapatkan kata-kata yang bersih dan bermakna.

Dokumen teks yang telah dibersihkan melalui tahapan-tahapan tersebut dapat dilanjutkan ke langkah berikutnya yaitu analisis.

2.2.3 Analisis Sentimen

Analisis sentimen atau *opinion mining* merupakan suatu proses mengolah data teks berupa bahasa alami, komputasi linguistik dan penambangan teks untuk dapat diperoleh informasi yang mudah dipahami. Manfaat dari analisis sentimen ialah untuk dapat memahami pendapat, sentimen, evaluasi, kritik, saran, sikap, emosi pembicara, serta persepsi seseorang atau masyarakat terhadap layanan, produk, organisasi, pribadi seseorang, suatu topik, peristiwa, serta hal-hal lainnya (Liu, 2010). Emosi yang terkandung dalam teks, dapat dianalisis kecenderungannya dalam menunjukkan perasaan marah, sedih, gembira, dan lainnya. Hasil dari analisis sentimen dapat dikelompokkan dalam sifat-sifat ujaran negatif, netral, maupun positif.

2.2.4 *Term Frequency Inverse Document Frequency (TF-IDF)*

Term Document Inverse Document Frequency (TF-IDF) merupakan metode untuk memberikan pembobotan pada setiap kata yang muncul dalam dokumen teks. Pembobotan dilakukan setelah dokumen melalui tahap pra proses teks, dimana selanjutnya akan dibuat model untuk mempermudah analisis. Kata-kata yang terdapat pada dokumen teks akan diubah

menjadi vektor lalu diberikan skor untuk setiap kemunculannya, lalu diolah dengan algoritma prediktif (Yudiarta, 2018). TF-IDF bertujuan untuk mencari jumlah kata pada dokumen teks (TF) setelah dikali dengan banyaknya tweet yang memiliki kata tersebut (IDF). Perhitungan TF-IDF dilakukan dengan cara integrasi metode *Term Frequency* (TF) dengan *Inverse Document Frequency* (IDF), mengikuti persamaan (2.1) dan (2.2).

$$w_{ij} = tf_{ij} \times idf \quad (2.1)$$

$$idf = \log \left(\frac{N}{df_j} \right) \quad (2.2)$$

Keterangan:

w_{ij} : Bobot dari kata ke-i pada artikel ke-j.

N : Jumlah seluruh tweet.

tf_{ij} : Jumlah kemunculan kata ke-i pada tweet ke-j.

df_j : Jumlah tweet ke-j yang mengandung kata ke-i.

2.2.5 *Naïve Bayes Classifier* (NBC)

Naïve Bayes Classifier (NBC) merupakan algoritma untuk melakukan klasifikasi yang didasarkan pada teori probabilitas bersyarat, dimana penggunaan bahasanya lebih sederhana. Algoritma *Naïve Bayes Classifier* dapat membuat asumsi bahwa dalam suatu data teks, kata yang muncul tidak mempengaruhi kata lainnya dan kata yang tidak muncul tidak mempengaruhi ketidakhadiran kata lainnya. Metode ini mengasumsikan bahwa kata-kata yang terdapat dalam teks bersifat independen dan tidak saling mempengaruhi. (Rocha, 2006). Persamaan (2.3) untuk *Naïve Bayes Classifier* yang mengacu pada teori probabilitas bersyarat adalah sebagai berikut.

$$P(a|v) = \frac{P(v|a)P(a)}{P(v)} \quad (2.3)$$

Keterangan :

$P(a|v)$: Probabilitas munculnya kata dalam persepsi masyarakat terhadap Kepolisian Negara Republik Indonesia berdasarkan sentimen

$P(v|a)$: Probabilitas munculnya kelas sentimen dalam persepsi masyarakat terhadap Kepolisian Negara Republik Indonesia

$P(v)$: Probabilitas munculnya sentimen

$P(a)$: Probabilitas dalam persepsi masyarakat terhadap Kepolisian Negara Republik Indonesia

Algoritma *Naïve Bayes Classifier* dapat digunakan untuk menemukan nilai probabilitas tertinggi untuk membuat klasifikasi yang paling tepat pada sebuah data uji (Feldman & Sanger, 2007). Penelitian ini menggunakan data yang diambil dari Twitter, dimana pada proses klasifikasi data tweet terdapat dua tahapan. Pertama adalah melatih data tweet yang telah diketahui kategorinya, sedangkan kedua adalah klasifikasi data tweet yang belum diketahui kategorinya (Falahah & Nur, 2015). Tiap kata dilambangkan dengan a_i dimana kata pertama adalah a_1 , kata kedua adalah a_2 , hingga kata ke-n atau a_n . V adalah himpunan kategori tweet, dimana saat proses klasifikasi, algoritma akan mencari probabilitas tertinggi dari semua kategori kata yang diuji dilambangkan dengan V_{MAP} mengikuti persamaan (2.4).

$$V_{MAP} = \arg \max P(v_j) \prod_i P(a_i|v_i) \quad (2.4)$$

$P(v_j)$ dihitung saat tahap pertama training data menggunakan persamaan (2.5)

$$P(v_j) = \frac{|dokumen_j|}{|training|} \quad (2.5)$$

$|dokumen_j|$ merupakan jumlah tweet dalam kategori sentimen ke- j dalam data *training*, sedangkan $|training|$ adalah jumlah tweet dalam sampel yang digunakan untuk *training*. $P(a_i|v_j)$ diperoleh dari persamaan (2.6).

$$P(a_i|v_j) = \frac{n_i+1}{|n+Kosakata|} \quad (2.6)$$

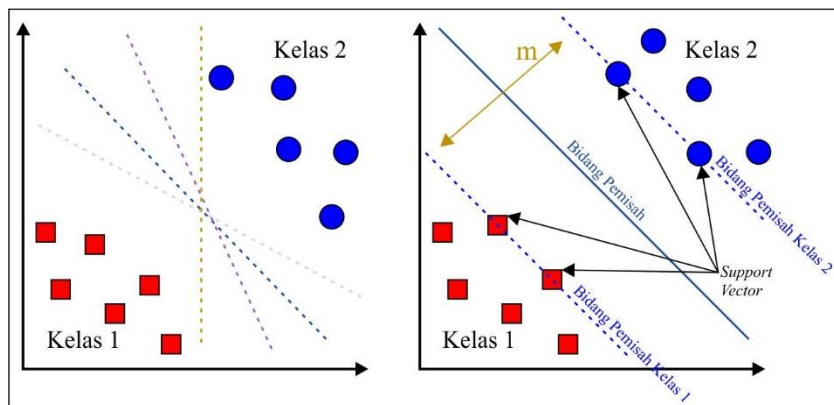
n_i merupakan jumlah kemunculan kata ke- i (a_i) dalam tweet berkategori v_j , serta n adalah jumlah banyaknya kata dalam tweet, sedangkan *Kosakata* adalah jumlah banyaknya kata dalam data *training*.

2.2.6 Support Vector Machine (SVM)

Support Vector Machine (SVM) termasuk dalam metode pembelajaran terbimbing (*supervised learning*) untuk mempelajari area pemisah antar kategori pada sebuah pengamatan, dimana terminologinya membahas tentang jarak atau margin antar kategori. Tiap kategori pada *Support Vector Machine* memiliki pengamatan dengan nilai variabel target yang sama (Williams, 2011). Kegunaan dari *Support Vector Machine* adalah untuk klasifikasi dan analisis regresi, dengan cara mencari *hyperplane* atau garis pembatas yang memisahkan antar kelas. *Hyperplane* terbaik atau support vector dapat ditemukan diantara fungsi yang jumlahnya tidak terbatas dengan metode ini (Widiastuti, 2007). Kelebihan dari metode *Support Vector Machine* adalah metode ini dapat bekerja dalam kondisi data apapun. Klasifikasi pada *Support Vector Machine* dapat bekerja pada kondisi linear (*Linearly Separable Data*) maupun non linear (*Non Linearly Separable Data*) yang dijelaskan sebagai berikut.

2.2.6.1 SVM pada *Linearly Separable Data*

Penerapan SVM pada *Linearly Separable Data* dimaksudkan untuk kondisi data yang dapat dipisahkan secara linier, sebagai contoh dimisalkan $x_i = \{x_i, x_{i+1}, \dots, x_n\}$ adalah set data sedangkan $y_i = \{+1, -1\}$ adalah label kategori untuk set data. Berikut ini adalah penggambaran dari *Linearly Separable Data* untuk SVM.



Gambar 2. 1 Alternatif Bidang Pemisah (Kiri) dan Bidang Pemisah Terbaik dengan Margin (m) Terbesar (Kanan)

Pada Gambar 2.1 kelas data pertama dan kedua dipisahkan oleh sepasang garis sejajar (linier), dimana data yang terletak pada bidang pembatas disebut *support vector*. Kelas pertama dibatasi oleh bidang pembatas pertama, sedangkan kelas kedua dibatasi oleh bidang pembatas kedua dan dapat dinyatakan dalam persamaan (2.7) dan (2.8).

$$x_i w + b \geq +1, \text{ untuk } y_i = +1 \quad (2.7)$$

$$x_i w + b \leq -1, \text{ untuk } y_i = -1 \quad (2.8)$$

w adalah normal bidang sedangkan b adalah konstanta yang disebut bias, jarak bidang pemisah yang tegak lurus dari titik pusat koordinat dirumuskan dalam persamaan (2.9).

$$\frac{|b|}{\|\mathbf{w}\|} \quad (2.9)$$

$\|\mathbf{w}\|$ adalah jarak Euclidean dari \mathbf{w} , sehingga dapat diperoleh nilai margin (jarak) bidang pembatas yaitu: $\frac{2}{\|\mathbf{w}\|}$. Nilai margin dapat dimaksimalkan dengan mengalikan b dan \mathbf{w} terhadap sebuah konstanta sehingga dihasilkan nilai margin dikalikan konstanta yang sama. Pengaturan skala batasan untuk persamaan (2.7) dan (2.8) dapat dipenuhi dengan mengatur ulang skala b dan \mathbf{w} karena memaksimalkan $\frac{1}{\|\mathbf{w}\|}$ sama dengan meminimumkan $\|\mathbf{w}\|^2$ mengikuti persamaan (2.10).

$$y_i(\mathbf{x}_i\mathbf{w} + b) - 1 \geq 0 \quad (2.10)$$

Rumusan untuk mencari bidang pemisah terbaik dengan nilai margin terbesar mengikuti persamaan optimasi batasan (2.11) berikut.

$$\min \frac{1}{2} \|\mathbf{w}\|^2 \quad (2.11)$$

Permasalahan tersebut dapat diselesaikan dengan persamaan lagrange dengan menggunakan lagrange multiplier sehingga didapatkan persamaan (2.12)

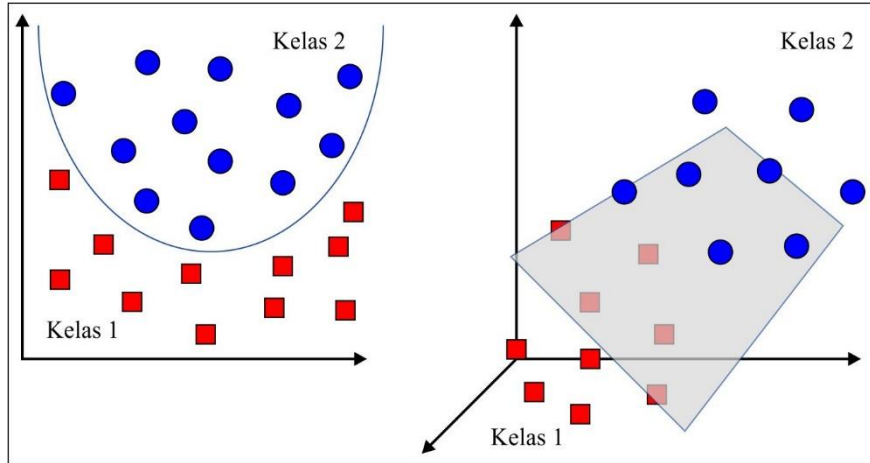
$$L_d = \sum_{m=1}^M a_m - \frac{1}{2} \sum_{m_1=1}^M \sum_{m_2=1}^M a_{m_1} a_{m_2} y_{m_1} y_{m_2} K(\mathbf{X}_{m_1}^T \mathbf{X}_{m_2}) \quad (2.12)$$

Persamaan tersebut digunakan untuk mencari nilai-nilai a_m (*support vector*) dengan mengoptimumkan L_d . L_d optimum diperoleh dengan mencari turunan sebagian dari L_d terhadap a , dan selanjutnya mencari nilai \mathbf{w} dan b dengan persamaan (2.13).

$$b = 1 - \mathbf{w}^T \mathbf{x} \quad \text{dimana; } \mathbf{w} = \sum_{i=1}^l a_i y_i \mathbf{x}_i \quad (2.13)$$

2.2.6.2 SVM pada *Non Linearly Separable Data*

Penerapan SVM pada *Non Linearly Separable Data* dimaksudkan untuk kondisi data yang tidak dapat dipisahkan secara linier, sehingga perlu modifikasi pada formulanya. Permasalahan tersebut dapat diselesaikan dengan melakukan transformasi dari ruang data menjadi ruang fitur berdimensi lebih tinggi, agar terbentuk pemisah yang linier. Ilustrasi untuk proses tersebut dijelaskan pada Gambar 2.2.



Gambar 2. 2 Ruang Data Dua Dimensi dengan Pemisah Non Linier (Kiri) dan Ruang Fitur Tiga Dimensi dengan Pemisah Linier (Kanan)

Proses mencari hyperplane optimal akan memerlukan data yang tidak terletak pada kelasnya (misclassification error) yang dilambangkan dengan ξ , mengikuti persamaan (2.14) dan (2.15).

$$\mathbf{x}_i \mathbf{w} + b \geq +1 - \xi_i, \text{ untuk kelas 1} \quad (2.14)$$

$$\mathbf{x}_i \mathbf{w} + b \leq -1 + \xi_i, \text{ untuk kelas 2} \quad (2.15)$$

Persamaan *Lagrange Multiplier* untuk *Non Linearly Separable Data* ditunjukkan pada persamaan (2.16).

$$L_d = \sum_{m=1}^M a_m - \frac{1}{2} \sum_{m_1=1}^M \sum_{m_2=1}^M a_{m_1} a_{m_2} y_{m_1} y_{m_2} K(\mathbf{X}_{m_1}, \mathbf{X}_{m_2}) \quad (2.16)$$

Data jenis tersebut dapat diselesaikan dengan metode kernel, dimana cara kerjanya adalah dengan melakukan transformasi data ke dalam dimensi ruang fitur hingga nantinya dipisahkan secara linier pada jarak fitur. Contohnya, misalkan ada suatu data x di jarak input pada jarak fitur dengan fungsi transformasi $x_k \rightarrow f(x_k)$ sehingga nilai $\mathbf{w} = \sum_{i=1}^n a_i y_i f(x_i)$ dan fungsi hasil yang telah dilatih ditunjukkan pada persamaan (2.17).

$$f(x_k) = \sum_{i=1}^n a_i y_i K(\mathbf{x}_{m_1}, \mathbf{x}_{m_2}) + b \quad (2.17)$$

Penggunaan persamaan (2.8) pada prakteknya, jarak fitur dapat memiliki dimensi yang tinggi dari vektor jarak input (Gunn, 1998). Kernel RBF memetakan hubungan tidak linear dengan parameter C , dimana C adalah parameter penentu besarnya nilai akibat kesalahan klasifikasi yang nilainya ditentukan pengguna. SVM menggunakan Kernel RBF dirumuskan dengan persamaan (2.18).

$$K(\mathbf{x}, \mathbf{c}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{c}\|^2}{\gamma}\right), \gamma > 0 \quad (2.18)$$

Fungsi Kernel RBF digunakan dalam penelitian ini karena mampu digunakan tanpa adanya pengetahuan terdahulu, dapat memetakan hubungan non linier, mampu mengatasi outlier karena berada pada rentang minus tak hingga sampai tak hingga ($-\infty$ sampai ∞), dan efektif menghindari overfitting dengan memilih nilai yang tepat untuk parameter C dan γ . Rentang yang dimiliki oleh fungsi kernel lainnya adalah -1 sampai 1 (Hsu, Chang, & Lin, 2003).

2.2.7 Holdout Validation

Holdout validation merupakan sebuah metode yang dapat digunakan untuk membagi data untuk *training* dan *testing*. Banyak peneliti yang menggunakan metode ini karena pembagian datanya dilakukan secara acak, dengan proporsi yang harus sedekat mungkin dengan proporsi pada populasi. Metode ini membagi data *training* dan data *testing* secara berulang, dimana setelah itu diambil rata-rata untuk analisis, yang kemudian proses ini disebut *repeated holdout* (Nugraha *et al.*, 2013).

	Data	
Iterasi 1	Test	Train
Iterasi 2	Train	Test
Iterasi 3	Test	Train
Iterasi 4	Train	Test
⋮	⋮	⋮
Iterasi-K	Test	Train

Gambar 2. 3 Pembagian Data menggunakan *Holdout Validation*

2.2.8 Ketepatan Klasifikasi

Ketepatan klasifikasi merupakan tahapan dalam menilai akurasi dari hasil klasifikasi yang telah dibuat, dalam pengukurannya diperlukan jumlah pada setiap kelas prediksi dan kelas aktual dengan ketentuan sebagai berikut:

- TP (*True Positive*) : Prediksi tepat jumlah tweet dengan sentimen positif dalam kelas positif.
- TN (*True Negative*) : Prediksi tepat jumlah tweet dengan sentimen negatif dalam kelas negatif.
- FP (*False Positive*) : Prediksi jumlah tweet dengan sentimen negatif dalam kelas positif.
- FN (*False Negative*) : Prediksi jumlah tweet dengan sentimen positif dalam kelas negatif.

Keempat ketentuan tersebut dapat diilustrasikan dalam Tabel 2.1.

Kelas Aktual	Kelas Prediksi	
	Positif	Negatif
Positif	TP	FN
Negatif	FP	TN

Kriteria yang biasa diperhatikan dalam mengukur ketepatan klasifikasi adalah akurasi, spesifikasi, dan sensitivitas (Hotho, Nurnberger, & Paass, 2005). Masing-masing dari kriteria tersebut dirumuskan dalam persamaan (2.19), (2.20), (2.21).

$$Akurasi = \frac{TN+TP}{TN+TP+FN+FP} \quad (2.19)$$

$$Spesifikasi = \frac{TN}{TN+FP} \quad (2.20)$$

$$Sensitivitas = \frac{TP}{TP+FN} \quad (2.21)$$

Pada data tidak seimbang digunakan *G-mean* (*geometric mean*) untuk mengukur ketepatan klasifikasi, yang merupakan rata-rata geometric nilai *recall* dari data dengan dua

kategori (Sun, Kamel, & Wang, 2006). Cara lain digunakan untuk mengukur ketepatan klasifikasi pada data tidak seimbang adalah nilai *Area Under Curve (AUC)* yang merupakan indikator performa kurva *Receiver Operating Characteristic (ROC)* untuk meringkas kinerja alat klasifikasi menjadi satu nilai dengan persamaan (2.22), (2.23) (Bekkar, Djemaa, & Alitouch, 2013).

$$G - mean = \sqrt{Sensitivitas \times Spesififikasi} \quad (2.22)$$

$$AUC = \frac{1}{2} (Sensitivitas + Spesififikasi) \quad (2.23)$$

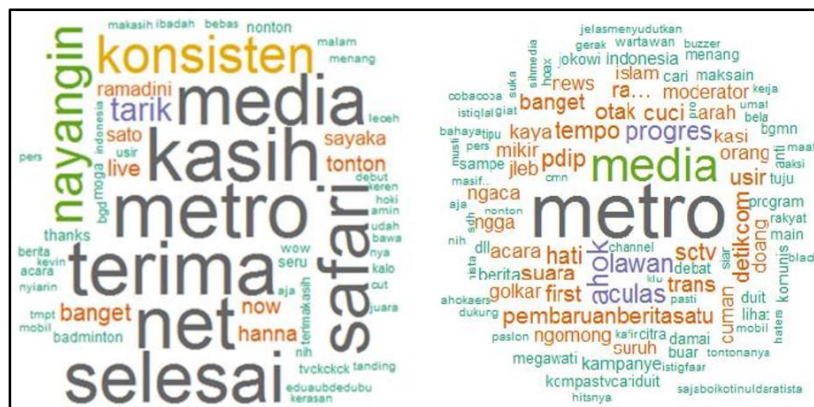
Skor yang muncul pada AUC dapat dikategorikan, dimana kategorinya ditunjukkan pada Tabel 2.2.

Tabel 2. 2 Kategori Skor AUC

Kategori	Skor
Sangat Baik	0,9 – 1
Baik	0,8 – 0,9
Cukup	0,7 – 0,8
Lemah	0,6 – 0,7
Gagal	0,5 – 0,6

2.2.9 Word Cloud

Word cloud merupakan salah satu cara untuk visualisasi data teks yang berupa representasi gambar dari sekumpulan data teks dengan melakukan plot tiap kata. Penempatan kata-kata dimulai dari yang paling sering muncul dengan ukuran gambar terbesar, hingga yang paling jarang muncul dengan ukuran makin kecil (Castella & Sutton, 2014). Berikut ini merupakan contoh *word cloud* yang ditunjukkan oleh Gambar 2.4.



Gambar 2. 4 Contoh *Word Cloud* sentimen positif (kiri) dan sentimen negatif (kanan)

Gambar 2.4 merupakan contoh word cloud tentang tweet masyarakat terhadap media Kompas TV dan Metro TV. Pada sentimen positif terdapat kata “terima kasih” muncul dominan karena masyarakat merasa tayangan Kompas TV dan Metro TV dinilai edukatif dan menghibur, sedangkan pada sentimen negatif banyak kata-kata mengandung unsur politik yang dianggap rentan mengundang perdebatan (Kurniawan, 2017).

2.3 Polisi

Polisi merupakan badan pemerintah yang bertugas memelihara keamanan dan ketertiban umum. Tugas tersebut dilaksanakan dengan menangkap orang yang melanggar hukum, patroli, menerima laporan masyarakat, dan sebagainya. Lingkup yang menjadi

tanggungjawab polisi dalam melaksanakan tugasnya antara lain, ekonomi, hukum, keagamaan, lalu lintas, militer, moral, negara, pamong praja, perairan (laut), sektor rahasia, dan kesusilaan (KBBI, 2021).

2.4 Twitter

Twitter merupakan sebuah situs jejaring yang beroperasi dibawah Twitter Inc., dimana situs ini bekerja sebagai media sosial berupa mikroblog sehingga memfasilitasi penggunaanya untuk saling berkirim dan membaca pesan tweet (Twitter, 2021). Pesan pada situs twitter disebut dengan tweet, dimana pengguna dapat mengirimkan tweet melalui akun yang telah dibuatnya. Tweet berisi maksimal 140 karakter yang dapat berisi apa saja sesuai dengan keinginan hati pengguna. Aplikasi Twitter juga hadir bagi masyarakat dan dapat diunduh khususnya untuk pengguna ponsel agar lebih mudah diakses selain melalui situs jejaring.

2.5 Citra Kepolisian

Databoks Indonesia merupakan sebuah lembaga survei yang mengeluarkan hasil pengamatan mengenai persentase kepercayaan masyarakat terhadap lembaga-lembaga di Indonesia. Tentara Nasional Indonesia menempati urutan pertama dengan persentase kepercayaan sebesar 92,2%, diikuti oleh Presiden Republik Indonesia dengan persentase kepercayaan sebesar 82,7%, sedangkan Kepolisian Negara Republik Indonesia ada di peringkat ketiga sebesar 74,1%. Kepolisian Negara Republik Indonesia mengalami penurunan paling tajam dibandingkan dengan semua lembaga yang disurvei sebesar 7,6% dari survei sebelumnya pada November 2021 yaitu sebesar 80,2%. Survei dilakukan terhadap delapan lembaga di Indonesia pada Desember 2021 (Databoks, 2022). Berdasarkan jurnal dari Jurusan Psikologi Universitas Negeri Jakarta diperoleh hasil analisis persepsi dan kepercayaan masyarakat terhadap kepolisian. Persepsi negatif lebih banyak daripada persepsi positif, dengan tingkat kepercayaan sedang. Terdapat pengaruh signifikan antara persepsi masyarakat dan kinerja kepolisian terhadap kepercayaan masyarakat kepada kepolisian, dimana model tersebut memiliki korelasi positif (Febrianti et al, 2013).

Halaman ini sengaja dikosongkan

BAB III METODOLOGI PENELITIAN

3.1 Metode Pengumpulan Data

Data yang digunakan dalam penelitian ini adalah data sekunder yang diambil dengan metode *crawling* pada platform Twitter, sehingga diperoleh data mentah yang tidak terstruktur. *Crawling* dilakukan dengan menggunakan aplikasi RapidMiner, dimana kata kunci yang digunakan adalah polisi dan polri. Data pada penelitian ini merupakan tweet yang muncul pada Januari 2022 sejumlah 3506 *tweet*, dimana setelah pra proses teks tersisa 2792 *tweet*.

3.2 Struktur Data dan Variabel Penelitian

Struktur data sebelum pra proses teks pada penelitian ini akan berupa tabel yang berisi data tweet sebagai variabel X dan klasifikasi sentimen tweet dalam “positif atau negatif” sebagai variabel Y. Contoh struktur data tweet sentimen masyarakat terhadap Kepolisian Negara Republik Indonesia sebelum pra proses yang diperoleh berdasarkan survei pendahuluan ditunjukkan pada Tabel 3.1.

Tabel 3. 1 Contoh Struktur Data

No	Tweet	Sentimen
1	Dua Oknum Polisi Aniaya Jurnalis Tempo Divonis 10 Bulan	negatif
2	Oknum Polisi Pukul Driver Ojol, Kapolres Bogor Minta Maaf	negatif
3	Motor ojol hilang & lapor polisi, dan dipukul oleh polisi krn gak ada uang pelicin Cepat2 Kapolsek Cileungsi Kompol Andri Alam bilang korban sdh ditangani	negatif
4	Aniaya Wartawan di Surabaya, Dua Oknum Polisi Hanya Divonis 10 Bulan Penjara	negatif
5	Suka iri sama polisi luar tuh, apa” langsung ditindak. Polisi indo? bbeehhh duit dulu bos baru jalan. (Cuma oknum sih tp kalau ngumpul kek lulusan snmptn)	negatif

Berdasarkan Tabel 3.1 data yang ditunjukkan dalam contoh sebelum pra proses, bila data telah melalui pra proses maka akan berbentuk kata dasar dari tiap *tweet*. Variabel yang digunakan dalam penelitian ini adalah data tweet masyarakat terhadap Kepolisian Negara Republik Indonesia yang sudah melalui tahap pra proses sebagai variabel X, yang kemudian dilakukan analisis sentimen berdasarkan kelas positif dan negatif sebagai variabel Y. Penjelasan variabel penelitian ini terdapat pada Tabel 3.2.

Tabel 3. 2 Variabel Penelitian

Variabel	Nama Variabel	Skala
Y	Kelas sentimen	Nominal
X	Kata-kata dari tweet setelah pra proses	Rasio

Pada Tabel 3.2 terdapat variabel Y yaitu kelas sentimen dengan skala nominal karena hanya berisi hasil negatif atau positif, sedangkan variabel X adalah data tweet setelah pra proses yang telah diberi bobot pada tahap *Term Frequency Inverse Document Frequency*.

3.3 Metode Penelitian

Penelitian ini menggunakan data yang diambil dengan metode *text mining*, dimana data dalam bentuk tidak terstruktur dan kualitatif. Data yang awalnya berupa teks tidak terstruktur akan diubah menjadi terstruktur dalam bentuk baris dan kolom, dengan tahap-tahap pra proses teks. Langkah selanjutnya adalah mengubah data teks yang kualitatif menjadi kuantitatif, atau berupa angka dengan metode TF dan TF-IDF. Skor dalam tiap kata yang dimiliki oleh data tersebut, membuat data dapat dilanjutkan ke tahap klasifikasi.

Metode yang digunakan untuk analisis dalam penelitian ini adalah klasifikasi menggunakan *Naïve Bayes Classifier* dan *Support Vector Machine*. Data yang berupa teks tidak terstruktur menyebabkan kelas positif dan negatif tidak dapat dipisahkan secara linier, sehingga pada *Support Vector Machine* digunakan metode *non linearly separable data* dengan kernel RBF. *Naïve Bayes Classifier* akan dibandingkan dengan *Support Vector Machine*, untuk mencari metode klasifikasi terbaik. Indikator ketepatan klasifikasi yang digunakan untuk membandingkan kedua metode tersebut adalah skor AUC, dimana skor AUC tertinggi akan dipilih menjadi metode klasifikasi terbaik untuk topik analisis sentimen masyarakat terhadap Kepolisian Negara Republik Indonesia. Hasil pelabelan kelas positif dan negatif akan divisualisasikan dalam bentuk *wordcloud*, untuk melihat kata-kata yang terbanyak muncul.

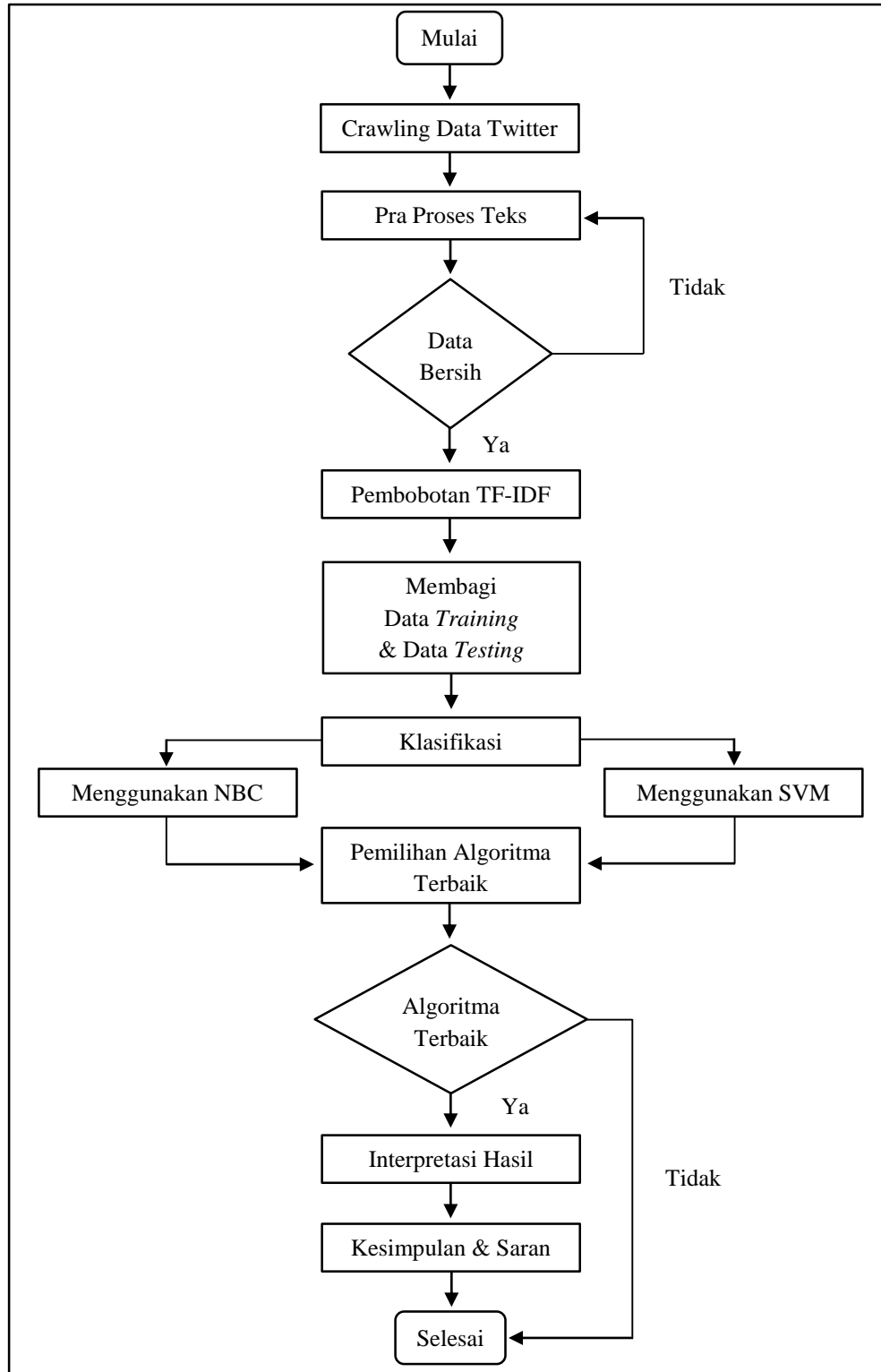
3.4 Langkah Analisis

Berikut adalah langkah analisis pada penelitian ini.

1. *Crawling* Data Twitter
 - a. Memasukkan kata kunci polisi dan polri.
 - b. Menyimpan data dalam bentuk *xlsx*.
2. Menyiapkan data *tweet*, daftar *stopwords*, kamus leksikon, dan kata dasar.
3. Melakukan pra proses teks.
 - a. Memberi label kelas sentimen pada data *tweet*.
 - b. Menghapus data *tweet* dengan kelas netral.
 - c. Membersihkan data *tweet* dari karakter-karakter yang tidak dibutuhkan.
4. Melakukan pembobotan TF-IDF.
 - a. Memberi bobot TF pada *term* dengan *sparse* 92%.
 - b. Melakukan transformasi hasil pembobotan TF menjadi TF-IDF mengikuti persamaan (2.1), dan (2.2).
5. Membagi *data training* dan *data testing* menggunakan *holdout validation* sebanyak masing-masing 10 kali.
6. Klasifikasi menggunakan *Naïve Bayes Classifier*.
 - a. Menggunakan hasil pembobotan TF untuk klasifikasi.
 - b. Dilakukan sebanyak 10 kali pada data *training*, dan 10 kali pada data *testing*.
 - c. Mencatat hasil ketepatan klasifikasi pada sensitivitas, spesifisitas, *G-Mean*, dan AUC.
7. Klasifikasi menggunakan *Support Vector Machine*.
 - a. Menggunakan hasil pembobotan TF-IDF untuk klasifikasi.
 - b. Menentukan parameter C dan gamma terbaik, untuk kernel RBF.
 - c. Dilakukan sebanyak 10 kali pada data *training*, dan 10 kali pada data *testing*.
 - d. Mencatat hasil ketepatan klasifikasi pada sensitivitas, spesifisitas, *G-Mean*, dan AUC.
8. Pemeriksaan ketepatan klasifikasi.
 - a. Menggunakan skor *G-Mean* dan AUC karena data tidak seimbang.
 - b. Memilih metode terbaik dari setiap indikator ketepatan klasifikasi.
9. Visualisasi *word cloud*.
10. Menarik kesimpulan dan saran.

3.5 Diagram Alir

Berikut merupakan diagram alir yang digunakan pada penelitian ini.



Gambar 3. 1 Diagram Alir

Halaman ini sengaja dikosongkan

BAB IV ANALISIS DAN PEMBAHASAN

Bab ini menjelaskan tentang hasil analisis dan pembahasan mengenai sentimen masyarakat terhadap Kepolisian Negara Republik Indonesia. Terdiri dari karakteristik data, hingga pemilihan metode klasifikasi terbaik antara *Naïve Bayes Classifier* dan *Support Vector Machine*. Tahap pra proses dan klasifikasi dilakukan dengan bantuan aplikasi RStudio, sedangkan TF-IDF menggunakan aplikasi Microsoft Excel, dan pencarian parameter untuk SVM menggunakan Python.

4.1 Pra Proses Data *Tweet*

Pra proses merupakan serangkaian tahapan yang dilakukan sehingga data siap untuk analisis. Tahapan-tahapan pra proses terdiri dari pemberian skor dan pembersihan data *tweet*. Langkah ini diperlukan karena penelitian ini menggunakan data tidak terstruktur, dan perlu diubah menjadi data terstruktur.

4.1.1 Pemberian Skor pada Data *Tweet*

Tahap ini dilakukan untuk menentukan tiap-tiap *tweet* mengandung skor total positif atau negatif. Skor diperoleh dari jumlah seluruh kata positif yang dikurangkan dengan jumlah seluruh kata negatif pada tiap *tweet*. Jika skor diatas 0, maka *tweet* masuk kelas positif. Jika skor sama dengan 0, maka *tweet* masuk kelas netral. Jika skor dibawah 0, maka *tweet* masuk kelas negatif. Hasil pemberian skor ditunjukkan pada Tabel 4.1 berikut.

Tabel 4. 1 Pemberian Skor

Skor	<i>Tweet</i>	Kelas
-3	wajib dipecat, polisi kok memperkosa polisi tuh mengayomi rakyat akwoawkowaaokwok https://t.co/WI0153uIr6	Negatif
-9	@Little_Alluka @txtdrberseragam Btw dia ga berbuat kak. Tapi si polisi memperkosa si wanita (mbak novia), karena awalnya si polisi ini mau dijodohin tapi dia lebih milih korban. Jadinya, mau nikah lari, dan berakhir memperkosa korban. Awalnya juga udah di	Negatif
-5	Ini videonya Dia pas lari ga kenapa napa, ga sakit sama sekali Udah sampe depan mobil, baru kesakitan Gila Tolong pak polisi yang begini dicituk agar yang lain ga ada yang jadi korban https://t.co/iJXRwEdngB	Negatif
-1	RT @rayestu: Gue jelas ga dukung ngatain kampung orang tempat jin buang anak, tapi sangat jelas kayak gituan doang kagak perlu diurus polisâ€¦ @txtdrberseragam Ga bersyukur sih....punya cewek cantik ga dijaga..pdhl tampang dia klo ga polisi ga tau bisa dpt cewek cantik apa kagak....klo lom bisa nikah resmi	Negatif
3	kan ya nikah siri dulu gt...ðŸ™,,	Positif
0	RT @Uki23: Walau berteman, saya kerap juga gak setuju dengan opini bang @FerdinandHaeen3. Tapi setidaknya belio gak pengecut saat dipanggilâ€¦ MantulllðŸ™ ðŸ– pak polisi Sadar BisnisðŸ™• Di Natura siapapun berhak suksesðŸ™• Ga memandang usia, ga memandang pendidikan, ga memandang profesi. Yg penting ada niat & kemauan. Bisnisnya ini sangat simple. Bisa dikerjakan oleh IRT, pegawai, mahasiswa/i/pelajar dsb.. Yuk minat WA 085772699134ðŸ™ ðŸ–½	Netral
3	https://t.co/OJmJ8D6O2R RT @BangkitSurabaya: LAUNCHING APLIKASI SAYANG WARGA	Positif
1	Kapolrestabes Surabaya Kombes Pol akhmad Yusep Gunawan didampingi Kasat Binmas dan Kâ€¦	Positif
-3	RT @L1ntingd4un: "Mahasiswa di Banjarmasin protes vonis rendah POLISI PEMERKOSA" Pembantai dilindungi, pemerkosa disayang, kurang bangsatâ€¦	Negatif
ï	ï	ï
0	@NalarNatural @DivHumas_Polri @CCICPolri Osang asing osang asing Mulu !	Netral

Tabel 4.1 menunjukkan data tweet yang telah diberi label positif, netral, dan negatif sesuai skor yang diperoleh. *Tweet* dengan kelas netral akan dibuang karena tidak membantu dalam pengambilan keputusan sesuai dengan batasan masalah. Proses ini dilakukan sebelum pembersihan data sehingga kata-kata dalam *tweet* masih dengan karakter yang lengkap untuk memberikan skor obyektif.

4.1.2 Pembersihan Data *Tweet*

Pembersihan data *tweet* dilakukan untuk menghilangkan karakter-karakter yang dianggap tidak perlu, serta mengondisikan data *tweet* ke dalam bentuk baris kolom terstruktur. Contoh langkah-langkah yang dilalui dalam proses pembersihan data *tweet* ditunjukkan pada Tabel 4.2.

Tabel 4. 2 Pembersihan Data *Tweet*

Tweet Asli	RT @SantorinisSun: Jiiiaah...cari masalah sendiri ini sih!ðŸŒŒâ€™,ï,â€” Harusnya hadir supaya situasi menjadi lebih kondusif https://t.co/X6mcKHb5au
Menghilangkan Link dan URL	RT: Jiiiaah...cari masalah sendiri ini sih!ðŸŒŒâ€™,ï,â€” Harusnya hadir supaya situasi menjadi lebih kondusif
Case Folding	rt jiiiaah cari masalah sendiri ini sih harusnya hadir supaya situasi menjadi lebih kondusif
Menghilangkan Stop Words	jiiiaah cari masalah sendiri harusnya hadir situasi lebih kondusif
Stemming	jiiiaah cari masalah sendiri harus hadir situasi lebih kondusif

Tabel 4.2 menunjukkan langkah demi langkah proses pembersihan data *tweet* hingga diperoleh *tweet* yang siap analisis. *Tweet* asli menunjukkan bahwa nama akun, *link*, tanda baca, imbuhan, kata hubung, dan karakter lainnya masih lengkap. Langkah kedua adalah menghilangkan link dan URL, dimana nama akun juga hilang. *Case folding* membuat karakter yang tersisa hanyalah huruf kecil, sehingga karakter selain huruf dihilangkan. Pada langkah menghilangkan *stop words* kata-kata yang tidak diperlukan hilang, termasuk kata hubung. *Stemming* menghapus imbuhan sehingga menyisakan kata dasar saja, hasil dari data setelah pembersihan ditunjukkan pada Tabel 4.3.

Tabel 4. 3 Data Bersih

<i>Tweet</i>
wajib pecat polisi perkosa polisi tuh ayom rakyat
dia buat kak polisi perkosa wanita mbak novia karena awal polisi mau jodoh dia lebih milih korban jadi mau nikah lari akhir perkosa korban awal juga janji bakal nikah kunjung nikah ini video dia pas lari kenapa napa sakit sama sekal sampe depan mobil baru sakit gila tolong pak polisi begini ciduk agar lain ada jadi korban
gue jelas dukung ngatain kampung orang tempat jin buang anak sangat jelas gitu doang kagak perlu urus polisi
syukur punya cewek cantik jaga pdhl tampang dia polisi tau bisa dpt cewek cantik apa kagak lom bisa nikah resmi nikah siri dulu
walau teman saya kerap juga setuju opini bang belio kecut saat panggil
⋮
bikin hoax lapor bengkek ngemis minta mediasi gmn kasus org sdh sampai dmn proses

Tabel 4.3 menunjukkan data *tweet* bersih dari karakter-karakter serta imbuhan yang dianggap tidak perlu. *Tweet* yang telah bersih akan melalui tahap *tokenizing* untuk membuat tiap kata menjadi independen, sehingga dapat terbaca jumlah kemunculannya. Jumlah kemunculan kata atau *Term Frequency (TF)* pada tiap *tweet* ditunjukkan pada Tabel 4.4.

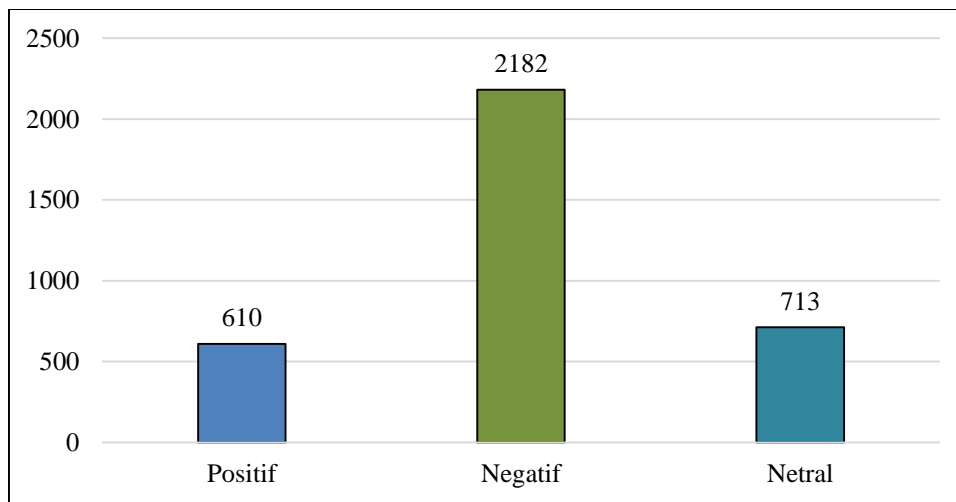
Tabel 4. 4 Term Frequency (TF)

Tweet ke	Kata							
	Perkosa	polisi	dan	Ini	Sama	Yang	...	Polri
1	1	2	0	0	0	0	...	0
2	2	2	1	1	0	0	...	0
3	0	1	0	1	1	3	...	0
4	0	0	0	0	0	0	...	0
5	0	1	0	0	0	0	...	0
6	0	0	0	0	0	0	...	0
7	0	1	0	1	0	0	...	0
8	0	0	1	0	0	0	...	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
2792	0	1	0	0	0	0	...	0

Tabel 4.4 menunjukkan bahwa telah diperoleh 2792 *tweet* setelah melalui tahap pra proses. Pada *tweet* ke-1 menunjukkan bahwa kata “perkosa” muncul sekali, kata “polisi” muncul dua kali, kata “dan” tidak muncul, dan seterusnya. Data yang semula tidak terstruktur, kini telah terstruktur dalam bentuk baris dan kolom sehingga siap untuk dilakukan klasifikasi.

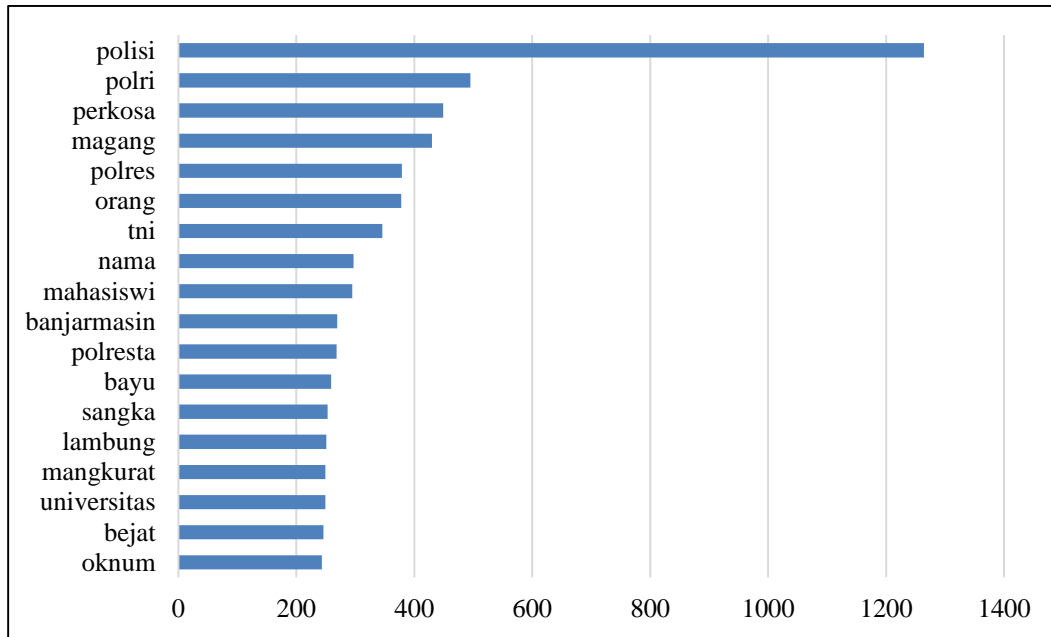
4.2 Karakteristik Data Sentimen Masyarakat Terhadap Kepolisian Negara Republik Indonesia

Karakteristik data digunakan untuk mendeskripsikan data *tweet* yang digunakan dalam penelitian ini, khususnya setelah melalui tahap pra proses. Jumlah *tweet* sesuai dengan kelas positif, negatif, dan netral ditunjukkan pada Gambar 4.1.



Gambar 4. 1 Jumlah *Tweet* per Kelas

Gambar 4.1 menunjukkan terdapat 610 *tweet* dalam kelas positif, 2182 *tweet* dalam kelas negatif, dan 713 *tweet* dalam kelas netral. *Tweet* dalam kelas netral akan dihapus sehingga menyisakan kelas positif dan negatif dengan total 2792 *tweet*. Jumlah *tweet* positif dan negatif menunjukkan data tak seimbang, dengan kelas negatif lebih banyak dari kelas positif. Frekuensi tiap kata yang muncul teratas dari keseluruhan *tweet* ditunjukkan pada Gambar 4.2.



Gambar 4. 2 Frekuensi Kata Teratas

Gambar 4.2 menunjukkan sebanyak 18 kata yang muncul terbanyak dari keseluruhan kata dalam data *tweet*. Kata “polisi” dan “polri” tentu menjadi dua teratas karena digunakan sebagai kata kunci, dengan jumlah kemunculan “polisi” sebanyak 1264 kata, dan “polri” sebanyak 495 kata. Kata “perkosa” menjadi non kata kunci yang terbanyak muncul, yaitu sebanyak 449 kata, sedangkan kata “oknum” berada di urutan ke 18 sebanyak 243 kata.

Frekuensi kemunculan kata juga dapat divisualisasikan dalam bentuk *wordcloud*, dimana urutan penulisan kata dari besar ke kecil sesuai dengan frekuensi terbanyak ke tersekit. *Wordcloud* data *tweet* sentimen masyarakat terhadap Kepolisian Negara Republik Indonesia secara keseluruhan ditunjukkan pada Gambar 4.2.



Gambar 4. 3 Wordcloud Data Tweet Keseluruhan

Gambar 4.3 menunjukkan kata-kata yang muncul terbanyak dari data *tweet* secara keseluruhan, dimana terlihat bahwa kata “polisi” dan “polri” sebagai dua kata yang terbanyak muncul secara berurutan, karena digunakan sebagai kata kunci saat *crawling*. Nampak beberapa kata yang menggambarkan kasus-kasus polisi sekitar bulan Januari 2022, seperti “perkosa”,

Negara Republik Indonesia akan dibagi menjadi data *training* dan data *testing*. Pembobotan data menggunakan *Term Frequency (TF)* seperti pada Tabel 4.3, dengan pembagian data *training* berjumlah 2234 *tweet* dan data *testing* berjumlah 558 *tweet*. Pembagian data *training* dan data *testing* dilakukan sebanyak 10 iterasi. Rata-rata dari ke-sepuluh iterasi percobaan ditunjukkan pada Tabel 4.5 menurut *confusion matrix*, serta skor sensitivitas, spesifisitas, *G-Mean*, dan *AUC*.

Tabel 4. 5 Ketepatan Klasifikasi NBC

Data	Confusion Matrix		Sensitivitas	Spesifisitas	G-Mean	AUC
	Prediksi	Aktual				
Training		Negatif	0.9928	0.2892	0.5358	0.6410
	Negatif	505				
	Positif	1241				
Testing		Negatif	0.9943	0.2780	0.5257	0.6361
	Negatif	121				
	Positif	315				

Tabel 4.5 menunjukkan bahwa pada ketepatan klasifikasi NBC, data *testing* memiliki skor lebih tinggi pada sensitivitas yaitu sebesar 0,9943, serta menjadi skor tertinggi. Ketepatan klasifikasi terendah pada data *testing* yaitu menggunakan spesifisitas, dengan skor 0,2780. Pada *confusion matrix* data *training*, data yang tepat terprediksi sebanyak 505 untuk kelas negatif dan 485 untuk kelas positif, sehingga totalnya 990 *tweet* tepat prediksi. *Confusion matrix* data *testing* menunjukkan 121 *tweet* pada kelas negatif, dan 121 *tweet* pada kelas positif tepat klasifikasi. Perhitungan manual sensitivitas, spesifisitas, *G-Mean*, dan *AUC* pada data *training* NBC mengikuti persamaan (2.20), (2.21), (2.22), dan (2.23), dimana hasilnya ditunjukkan pada persamaan (4.1), (4.2), (4.3), dan (4.4).

$$Sensitivitas = \frac{485}{485+4} = 0,9928 \quad (4.1)$$

$$Spesifisitas = \frac{505}{505+1241} = 0,0289 \quad (4.2)$$

$$G - mean = \sqrt{0,9928 \times 0,0289} = 0,5358 \quad (4.3)$$

$$AUC = \frac{1}{2}(0,9928 + 0,0289) = 0,6410 \quad (4.4)$$

Perhitungan manual untuk data *testing* NBC juga mengikuti persamaan tersebut. Proporsi untuk data *tweet* dalam tiap kelas untuk data *training* dan data *testing* ditunjukkan pada Tabel 4.6.

Tabel 4. 6 Proporsi *Tweet*

	Negatif	Positif
Training	0.781558	0.218442
Testing	0.781362	0.218638

Tabel 4.6 menunjukkan rincian proporsi data *tweet* dengan kelas negatif dibandingkan dengan kelas positif. Proporsi kelas positif dan negatif pada *tweet* sentimen masyarakat terhadap Kepolisian Negara Republik Indonesia menunjukkan bahwa sentimen negatif banyak muncul di Twitter. Perbandingan kelas negatif mendekati 80% dan kelas positif mendekati 20%.

4.4 Klasifikasi menggunakan *Support Vector Machine*

Support Vector Machine (SVM) merupakan salah satu metode klasifikasi dalam penelitian ini, dimana pembobotan katanya menggunakan *Term Frequency (TF)* dengan *Inverse Document Frequency (IDF)*. Langkah yang dilakukan adalah dengan mengintegrasikan pembobotan pada Tabel 4.4, dengan persamaan 2.2 sehingga diperoleh bentuk pembobotan kata baru yang ditunjukkan pada Tabel 4.7.

Tabel 4. 7 TF-IDF

tweet ke	Perkosa	Polisi	Orang	banjarmasin	Oknum	...	Tni	Klasifikasi
1	0.7936691	0.4477733	0	0	0	...	0	Negatif
2	1.0325873	0.4477733	0	0	0	...	0	Negatif
3	0	0.3441683	0	0	0	...	0	Negatif
4	0	0	0.8684236	0	0	...	0	Negatif
5	0	0.3441683	0	0	0	...	0	Positif
6	0	0.3441683	0	0	0	...	0	Positif
7	0	0	0	0	0	...	0	Positif
8	0	0.3441683	0	1.0161631	0	...	0	Negatif
9	0	0.3441683	0	0	0	...	0	Positif
10	0	0.3441683	0	0	1.06030914	...	0	Negatif
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
2792	0	0	0	0	0	...	0	Negatif

Tabel 4.7 menunjukkan hasil pembobotan menggunakan TF-IDF, dimana bobot yang muncul dapat berupa angka desimal. Pada penelitian sebelumnya menurut Sub Bab 2.1, metode TF-IDF merupakan pembobotan terbaik untuk digunakan dalam klasifikasi SVM. Data yang semula berupa teks telah dikonversi menjadi angka, sehingga dapat dilanjutkan ke tahap selanjutnya yaitu pembagian data *training* dan *data testing*. Pembagian data *training* berjumlah 2234 *tweet* dan data *testing* berjumlah 558 *tweet*. Pembagian data *training* dan data *testing* dilakukan sebanyak 10 iterasi.

Metode SVM yang digunakan adalah adalah kernel *Radial Basis Function (RBF)* sebagai kernel terbaik menurut penelitian sebelumnya, dan juga karena data teks termasuk non linier. Kernel RBF menggunakan parameter C dan gamma, sehingga perlu diketahui nilai C dan gamma terbaik untuk klasifikasi. Pencarian nilai C dan gamma terbaik dilakukan dengan bahasa pemrograman Python menggunakan bantuan aplikasi Google Collab. Hasil dari pencarian parameter C dan gamma terbaik ditunjukkan pada Gambar 4.6.

```
# Tuning hyper-parameters for recall

Best parameters set found on development set:

{'C': 1000, 'gamma': 1, 'kernel': 'rbf'}
```

Gambar 4. 6 Hasil Parameter C dan Gamma Terbaik

Gambar 4.6 menunjukkan bahwa diperoleh parameter C terbaik sebesar 1000, dan parameter gamma terbaik sebesar 1. Kedua parameter tersebut akan dipakai dalam melakukan klasifikasi SVM Kernel RBF dalam 10 iterasi percobaan. Rata-rata dari ke-sepuluh iterasi percobaan tersebut ditunjukkan pada Tabel 4.8 menurut *confusion matrix*, serta skor sensitivitas, spesifisitas, *G-Mean*, dan AUC.

skor AUC sebesar 0,6410 yang lebih unggul daripada SVM, sehingga NBC terpilih menjadi metode klasifikasi terbaik untuk data *training*. Ketepatan klasifikasi data *testing* untuk NBC dan SVM ditunjukkan pada Tabel 4.10.

Tabel 4. 10 Ketepatan Klasifikasi Data *Testing* NBC dan SVM

Data	Confusion Matrix		Sensitivitas	Spesifisitas	G-Mean	AUC
NBC	Prediksi	Aktual		0.9943	0.2780	0.5257
		Negatif	Positif			
	Negatif	121	1			
Positif	315	121				
SVM	Prediksi	Aktual		0.3008	0.9232	0.5270
		Negatif	Positif			
	Negatif	397	85			
Positif	33	37				

Tabel 4.10 menunjukkan bahwa pada *confusion matrix* data *testing* NBC unggul pada sensitivitas dan AUC, sementara SVM unggul pada spesifisitas dan G-Mean. NBC memiliki skor AUC sebesar 0,6361 yang lebih unggul daripada SVM, sehingga NBC terpilih menjadi metode klasifikasi terbaik untuk data *testing*. NBC yang memperoleh skor AUC sebesar 0,6410 untuk data *training* dan 0,6361 untuk data *testing*, masuk dalam kategori klasifikasi lemah merujuk pada Tabel 2.2.

Halaman ini sengaja dikosongkan

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan dari analisis sentimen masyarakat terhadap Kepolisian Negara Republik Indonesia menggunakan *Naïve Bayes Classifier* dan *Support Vector Machine* adalah sebagai berikut.

1. Persentase sentimen negatif dibandingkan dengan positif dari tweet masyarakat terhadap Kepolisian Negara Republik Indonesia adalah mendekati 80% dibandingkan 20%, dimana sentimen negatif jauh lebih banyak. Kata-kata yang banyak muncul pada *tweet* bersentimen positif adalah “mohon”, “kapolri”, “tertib”, ”salah”, dan “hukum”. Pada *tweet* bersentimen negatif kata-kata yang banyak muncul adalah “perkosa”, “magang”, “polres”, “tni”, dan “mahasiswa”.
2. NBC lebih unggul daripada SVM dengan skor AUC sebesar 0,6410 dan 0,6361 pada ketepatan klasifikasi data *training* dan data *testing*. Skor AUC sebesar 0,6410 dan 0,6361 masuk dalam kategori klasifikasi lemah.

5.2 Saran

Berdasarkan sentimen negatif yang diperoleh sebesar 78,1%, sebaiknya Kepolisian Negara Republik Indonesia melakukan pembinaan pada anggota yang melanggar. Perbaikan pada proses rekrutmen juga diperlukan untuk mendapatkan anggota terbaik, serta lebih proaktif terhadap kebutuhan masyarakat. Upaya-upaya tersebut diharapkan mampu meningkatkan citra kepolisian di masyarakat. Penelitian selanjutnya diharapkan dapat mencoba menggunakan metode klasifikasi selain NBC dan SVM, karena pada penelitian ini diperoleh klasifikasi lemah.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- BBC. (2020). *Indonesia*. <https://www.bbc.com/indonesia/indonesia-54611952>. [diakses 1 Juli 2022]
- BBC. (2021). *Indonesia*. <https://www.bbc.com/indonesia/indonesia-58911141>. [diakses 1 Juli 2022]
- Bekkar, M., Djemaa, H. K., & Alitouch, T. A. (2013). Evaluation Measure for Models Assesment over Imbalanced Data Sets. *Journal of Information Engineering and Aplications*, 3, 27- 38.
- Berry, M. W., & Kogan, J. (2010). *Text Mining Application and Theory*. United Kingdom: WILEY.
- Castella, Quim & Sutton, Charles. (2014). *Word Storm: Multiples of Word Clouds for Visual Comparison of Documents*.
- CNN. (2021). *Hukum Kriminal*. <https://www.cnnindonesia.com/nasional/20211015061441-12-708062/daftar-panjang-tindakan-represif-dan-kekerasan-polisi>. [diakses 1 Juli 2022]
- Databoks. (2022). *Katadata*. <https://databoks.katadata.co.id/datapublish/2022/01/10/survei-indikator-tni-jadi-institusi-paling-dipercaya-masyarakat>. [diakses 1 Juli 2022]
- Falahah & Nur, D. D. A. (2015). Pengembangan Aplikasi Sentiment Analysis Menggunakan Metode Naïve Bayes. *Seminar Nasional Sistem Informasi Indonesia*, 335-340
- Febriyanti, L., & Maulana, H. (2013). Pengaruh Persepsi Masyarakat pada Kinerja Kepolisian terhadap Kepercayaan pada Kepolisian. *Jurnal Penelitian dan Pengukuran Psikologi*, Vol. 2, No. 1
- Feldman, R., & Sanger, J. (2007). *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. New York: Cambridge University Press.
- Gokgoz, E., & Subasi, A. (2015). Comparison of Decision Tree Algorithms for EMG Signal Classification Using DWT. *Biomedical Signal Processing and Control*, 18, 138-144.
- Gunn, S. R. (1998). *Support Vector Machine for Classification and Regression*. Southampton: University of Southampton.
- Hotho, A., Nurnberger, A., & Paass, G. (2005). *A Brief Survey of Text Mining*. Kassel: University of Kassel.
- Hsu, C.-W., Chang, C.-C., dan Lin, C.-J. (2003). *A Practical Guide to Support Vector Classification*. Taiwan: National Taiwan University.
- Kompas. (2022). *Indonesia*. <https://www.kompas.com/tren/read/2022/04/30/161500865/divonis-2-tahun-penjara-berikut-perjalanan-kasus-bripda-randy-yang-terjerat>. [diakses 1 Juli 2022]
- Kurniawan, T. (2017). *Implementasi Text Mining Pada Analisis Sentimen Pengguna Twitter Terhadap Media Mainstream Menggunakan Naïve Bayes Classifier dan Support Vector Machine*. Tugas Akhir: Institut Teknologi Sepuluh Nopember Surabaya.
- Liu, B. (2010). *Handbook of Natural Lenguage Processing 2nd Edition*. Boca Raton: CRC Press.
- Nomleni, Petrix, (2015). *Sentiment Analysis menggunakan Support Vector Machine (SVM)*. Tesis. Institut Teknologi Sepuluh Nopember Surabaya.
- Nugraha, Praditio Aditya., Saptono, Ristu., Sulistyoo., Meiyanto Eko. (2013). Perbandingan Metode Probabilistik *Naïve Bayesian Classifier* dan Jaringan Syaraf Tiruan *Learning Vector Quantization* dalam Kasus Klasifikasi Penyakit Kandungan. *Jurnal ITSMART*, Vol.2 No.2, 2301 – 7201.
- Rocha, A. de R., (2006). *Naive Bayes Classifier Teaching Material*. , pp.1–9.
- Syakuro, Abdan, (2017). *Analisis Sentimen Masyarakat terhadap E-Commerce pada Media Sosial menggunakan Metode Naïve Bayes Classifier (NBC) dengan Seleksi Fitur Information Gain (IG)*. Skripsi. Universitas Islam Negeri Maulana Malik Ibrahim Malang.

- Twitter. (2021). *Twitter Support*. [http:// support.twitter.com/](http://support.twitter.com/). [diakses 7 Desember 2021]
- Universitas Brawijaya. (2022). *Fakultas Hukum*. <https://hukum.ub.ac.id/cermin-ketidakprofesionalan-polisi-sering-abai-laporan-warga/>. [diakses 1 Juli 2022]
- Weiss, S. M. (2010). *Text Mining: Predictive Methods for Analyzing Unstructural Information*. New York: Springer.
- Wibowo, A. S. (2018). Analisis sentimen publik pada program kesehatan masyarakat menggunakan twitter opinion mining. *Seminar Nasional Informatika Medis*, 1-21.
- Widiastuti, D., (2007). Analisa Perbandingan Algoritma SVM, Naïve Bayes, dan Decision Tree dalam Mengklasifikasikan Serangan (Attack) pada Sistem Pendeteksi Intrusi. *Skripsi. Jurusan Sistem Informasi Universitas Gunadarma*, pp.1–8.
- Williams, Graham. (2011). *Data Mining with Rattle and R: The Art of Excavating Data for Knowledge Discovery*. New York: Springer.
- Yudiarta, N. G., Sudarma, M., dan Ariastina, W. G. (2018). Penerapan Metode Clustering Text Mining Untuk Pengelompokan Berita Pada Unstructured Textual Data. *Majalah Ilmiah Teknologi Elektro*, 17(3), 339. <https://doi.org/10.24843/mite.2018.v17i03.p06>

LAMPIRAN

Lampiran 1 Syntax R Pelabelan Kelas Sentimen

```
install.packages("devtools")
library(devtools)
install_github("nurandi/katadasaR")
#LABELING KELAS
#Analisis Sentimen
library(ROAuth)
library(RCurl)

Datasantimen <- tweet_df
pos = scan('positif.txt',
          what='character', comment.char=;)
neg = scan('negatif.txt',
          what='character', comment.char=;)
pos.words = c(pos)
neg.words = c(neg)
score.sentiment = function(Datasantimen, pos.words, neg.words, .progress='none')
{
  require(plyr)
  require(stringr)

  # we got a vector of sentences. plyr will handle a list
  # or a vector as an "l" for us
  # we want a simple array of scores back, so we use
  # "l" + "a" + "ply" = "lapply":
  scores = lapply(Datasantimen, function(sentence, pos.words, neg.words) {
    #gsub() function replaces all matches of a string
    # clean up sentences with R's regex-driven global substitute, gsub():
    sentence = gsub('[:punct:]', "", sentence)
    sentence = gsub('[:cntrl:]', "", sentence)
    sentence = gsub('\\d+', "", sentence) #angka
    # and convert to lower case:
    sentence = tolower(sentence)

    # split into words. str_split is in the stringr package
    word.list = str_split(sentence, "\\s+")
    # sometimes a list() is one level of hierarchy too much
    words = unlist(word.list)

    # compare our words to the dictionaries of positive & negative terms
    pos.matches = match(words, pos.words)
    neg.matches = match(words, neg.words)
  })
}
```

Lampiran 1 Syntax R Pelabelan Kelas Sentimen (Lanjutan)

```
# match() returns the position of the matched term or NA
# we just want a TRUE/FALSE:
pos.matches = !is.na(pos.matches)
neg.matches = !is.na(neg.matches)

# and conveniently enough, TRUE/FALSE will be treated as 1/0 by sum():
score = sum(pos.matches) - sum(neg.matches)

return(score)
}, pos.words, neg.words, .progress=.progress )

scores.df = data.frame(score=scores, text=Datasantimen)
return(scores.df)
}

sentimen2class = score.sentiment(Datasantimen$text, pos.words, neg.words)
View(sentimen2class)

sum(sentimen2class$score)
sentimen2class$klasifikasi=ifelse(sentimen2class$score> 0,"Positif",ifelse
(sentimen2class$score<0,"Negatif", "Netral"))

sentimen2class$klasifikasi
View(sentimen2class)
write.csv(sentimen2class, "sentimen.csv", row.names = TRUE)
```

Lampiran 2 Syntax R Pra Proses Data

```
library(NLP)
library(tm)
library(stringr)
library(caret)
library(dplyr)
library(katadasaR)
library(tau)
library(parallel)
library(tokenizers)
library(wordcloud)

#merubah file .csv tadi ke dalam corpus
corpusdata <- Corpus(VectorSource(tweet_df$text))
#mengecek atau melihat bentuk corpus dataset dokumen ke 1 sampai ke 100
inspect(corpusdata[1:100])
#mengubah semua huruf kapital menjadi huruf kecil

data_casfolding <- tm_map(corpusdata, content_transformer(tolower))
#cek data_casfolding
inspect(data_casfolding[1:100])

#Menghapus url pada dokumen
removeURL <- function(x)gsub("http[^[:space:]]*", "", x)
dataURL <- tm_map(data_casfolding, content_transformer(removeURL))
#Menghapus url pada data ke 1 sd 100
inspect(dataURL[1:100])

#Menghapus mention pada dokumen
remove.mention <- function(x) gsub ("@\\S+", "", x)
data_mention <- tm_map(dataURL, remove.mention)
#Menghapus mention pada data ke 1 sd 100
inspect(data_mention[1:100])

#Menghapus hashtag
removehashtag <- function(x)gsub("#\\S+", "", x)
datahashtag <- tm_map(data_mention, removehashtag)
#Menghapus Hastag pada data ke 1 sd 100
inspect(datahashtag[1:100])

#Cleaning Punctuation
datapunctuation <- tm_map(datahashtag, content_transformer(removePunctuation))
inspect(datapunctuation[1:100])
```

Lampiran 2. Syntax R Pra Proses Data (Lanjutan)

```
#Cleaning Number
datanonumber <- tm_map(datapunctuation, content_transformer(removeNumbers))
inspect(datanonumber[1:100])

#stemming atau menghapus imbuhan sehingga kata menjadi kata dasar
stemming <- function(text, mc.cores=1){
  stem_string <- function(str){
    str <- tokenize(x=str)
    str <- sapply(str, katadasaR)
    str <- paste(str, collapse = "")
    return(str)
  }
  x <- mclapply(X=text, FUN=stem_string, mc.cores=mc.cores)
}
data_stemming <- tm_map(datanonumber, stemming)
inspect(data_stemming[1:100])
#Simpan stemming
databersih <- data.frame(text=unlist(sapply(data_stemming, '[')),stringsAsFactors = FALSE)
write.csv(databersih, file = "databersih.csv")

#Memanggil Databersih
databersih <- read.csv(file.choose(""), header = TRUE, stringsAsFactors = TRUE, sep=";")
corpusdatabersih <- Corpus(VectorSource(databersih$text))

#menghapus kata-kata tidak penting
cStopwordID <- readLines("stop_words_ind.txt")
data_stopword <- tm_map(corpusdatabersih, removeWords, cStopwordID)
inspect(data_stopword[1:100])

#menghapus spasi berlebih
data_whitespace <- tm_map(data_stopword, stripWhitespace)
inspect(data_whitespace[1:100])
#menyimpan data spasi
databersih2 <- data.frame(text=unlist(sapply(data_whitespace, '[')),stringsAsFactors = FALSE)
write.csv(databersih2, file = "databersih2.csv")

#Tokenizer memilah/ memecah tweet kedalam bentuk terms/kata
corpustext <- Corpus(VectorSource(databersih2$text))
inspect(corpustext[1:100])
text = corpustext
strsplit_space_tokenizer <- function(x)
  unlist(strsplit(as.character(x), "[[:space:]]+"))
strsplit_space_tokenizer(text)
```

Lampiran 3 Syntax R TF

```
#TermFrequency
dabersih3 <- data.frame(text=unlist(sapply(text,'[]'),stringsAsFactors = FALSE))
write.csv(dabersih3, file = "dabersih3.csv")

#Memanggil Dabersih 3
dabersih3<-read.csv(file.choose(""), header = TRUE, stringsAsFactors = TRUE, sep=";")

corpusdabersih3 <- Corpus(VectorSource(dabersih3$text))

tdm <- TermDocumentMatrix(corpusdabersih3, control = list(minWordLength=c(1,Inf)))
tdm
t <- removeSparseTerms(tdm, sparse = 0.92)
t
m <- as.matrix(t)
m
freq <- rowSums(m)
write.csv(m, "mdata.csv", row.names = TRUE)
write.csv(freq, "frekuensikata.csv", row.names = TRUE)

#PembagianData
#Memanggil DataNBC
DataNBC <- read.csv(file.choose(""), header = TRUE, stringsAsFactors = TRUE, sep=",")

#Variabel terbentuk
sentimen2class = rbind(DataNBC[which(DataNBC$klasifikasi == "Negatif"),],
                      DataNBC[which(DataNBC$klasifikasi == "Positif"),])
sentimen2class[,19] = factor(sentimen2class[,19])
str(sentimen2class)

#Persiapan Working Directory
Negatif = sentimen2class[which(sentimen2class$klasifikasi == "Negatif"),]
Positif = sentimen2class[which(sentimen2class$klasifikasi == "Positif"),]

library(caTools)
```

Lampiran 4 Syntax R TF-IDF

```
#TF IDF
Mengubah hasil pembobotan TF menjadi TF-IDF secara manual menggunakan aplikasi
Microsoft Excel, lalu file disimpan dengan nama "DataSVM"

#Memanggil DataSVM
DataSVM <- read.csv(file.choose(""), header = TRUE,
                    stringsAsFactors = TRUE, sep=",")
#Variabel terbentuk
sentimen2classSVM = rbind(DataSVM[which(DataSVM$klasifikasi
                                       == "Negatif"),],
                          DataSVM[which(DataSVM$klasifikasi ==
                                       "Positif"),])
sentimen2classSVM[,19] = factor(sentimen2classSVM[,19])
str(sentimen2classSVM)
#Persiapan Working Directory
Negatif = sentimen2classSVM[which(sentimen2classSVM$klasifikasi ==
                                  "Negatif"),]
Positif = sentimen2classSVM[which(sentimen2classSVM$klasifikasi ==
                                  "Positif"),]
library(caTools)
```


Lampiran 5 Syntax R Holdout Validation

```
#cross fold validation 1
set.seed(1232342)
split= sample.split(sentimen2class$klasifikasi, SplitRatio = 0.8)
training_set1= subset(sentimen2class, split == TRUE)
test_set1= subset(sentimen2class, split == FALSE)
write.csv(training_set1, "sentimen2class.train.1.csv", row.names = FALSE)
write.csv(test_set1, "sentimen2class.test.1.csv", row.names = FALSE)

#cross fold validation 2
set.seed(12414141)
split= sample.split(sentimen2class$klasifikasi, SplitRatio = 0.8)
training_set2 = subset(sentimen2class, split == TRUE)
test_set2 = subset(sentimen2class, split == FALSE)
write.csv(training_set2, "sentimen2class.train.2.csv", row.names =
FALSE)
write.csv(test_set2, "sentimen2class.test.2.csv", row.names = FALSE)

#cross fold validation 3
set.seed(53453535)
split= sample.split(sentimen2class$klasifikasi, SplitRatio = 0.8)
training_set3 = subset(sentimen2class, split == TRUE)
test_set3 = subset(sentimen2class, split == FALSE)
write.csv(training_set3, "sentimen2class.train.3.csv", row.names =
FALSE)
write.csv(test_set3, "sentimen2class.test.3.csv", row.names = FALSE)

#cross fold validation 4
set.seed(343425)
split= sample.split(sentimen2class$klasifikasi, SplitRatio = 0.8)
training_set4 = subset(sentimen2class, split == TRUE)
test_set4 = subset(sentimen2class, split == FALSE)
write.csv(training_set4, "sentimen2class.train.4.csv", row.names =
FALSE)
write.csv(test_set4, "sentimen2class.test.4.csv", row.names = FALSE)

#cross fold validation 5
set.seed(2525225)
split= sample.split(sentimen2class$klasifikasi, SplitRatio = 0.8)
training_set5= subset(sentimen2class, split == TRUE)
test_set5= subset(sentimen2class, split == FALSE)
write.csv(training_set5, "sentimen2class.train.5.csv", row.names =
FALSE)
write.csv(test_set5, "sentimen2class.test.5.csv", row.names = FALSE)
```

Lampiran 5. Syntax R Holdout Validation (Lanjutan)

```
#cross fold validation 6
set.seed(23131345)
split= sample.split(sentimen2class$klasifikasi, SplitRatio = 0.8)
training_set6= subset(sentimen2class, split == TRUE)
test_set6= subset(sentimen2class, split == FALSE)
write.csv(training_set6, "sentimen2class.train.6.csv", row.names =
          FALSE)
write.csv(test_set6, "sentimen2class.test.6.csv", row.names = FALSE)

#cross fold validation 7
set.seed(1212877283)
split= sample.split(sentimen2class$klasifikasi, SplitRatio = 0.8)
training_set7= subset(sentimen2class, split == TRUE)
test_set7= subset(sentimen2class, split == FALSE)
write.csv(training_set7, "sentimen2class.train.7.csv", row.names =
          FALSE)
write.csv(test_set7, "sentimen2class.test.7.csv", row.names = FALSE)

#cross fold validation 8
set.seed(5324325)
split= sample.split(sentimen2class$klasifikasi, SplitRatio = 0.8)
training_set8= subset(sentimen2class, split == TRUE)
test_set8= subset(sentimen2class, split == FALSE)
write.csv(training_set8, "sentimen2class.train.8.csv", row.names =
          FALSE)
write.csv(test_set8, "sentimen2class.test.8.csv", row.names = FALSE)

#cross fold validation 9
set.seed(34346982)
split= sample.split(sentimen2class$klasifikasi, SplitRatio = 0.8)
training_set9= subset(sentimen2class, split == TRUE)
test_set9= subset(sentimen2class, split == FALSE)
write.csv(training_set9, "sentimen2class.train.9.csv", row.names =
          FALSE)
write.csv(test_set9, "sentimen2class.test.9.csv", row.names = FALSE)

#cross fold validation 10
set.seed(25919213)
split= sample.split(sentimen2class$klasifikasi, SplitRatio = 0.8)
training_set10= subset(sentimen2class, split == TRUE)
test_set10= subset(sentimen2class, split == FALSE)
write.csv(training_set10, "sentimen2class.train.10.csv", row.names =
          FALSE)
write.csv(test_set10, "sentimen2class.test.10.csv", row.names = FALSE)
```

Lampiran 6 Syntax R *Naïve Bayes Classifier*

```
#NBC TRAINING
library(e1071)
library(caret)

#membaca data training & data training
data.train1 = training_set10
data.train2 = training_set10

str(data.train2)
str(data.train1)

#membuat model
modell = naiveBayes(as.factor(klasifikasi)~., data.train1)
modell

#melakukan prediksi
predict_result = predict(modell, data.train2[,-19])
predict_result

#menghitung kinerja
performance.value = confusionMatrix(table(predict_result, data.train2$klasifikasi), positive =
"Positif")
print(performance.value)

comparison_result = cbind(prediction = as.character(predict_result),
                           actual = as.character(data.train2[,19]))
comparison_result

write.csv(comparison_result, "cr_train10.csv", row.names = FALSE)

#menghitung kinerja
library(ROCR)
roc.prediction = prediction(as.numeric(as.factor(predict_result)),
                           as.numeric(as.factor(data.train2[,19])))
roc.tpr.fpr = performance(roc.prediction, "tpr", "fpr")
roc.auc = performance(roc.prediction, "auc")
plot(roc.tpr.fpr, col="red", lty=3)
abline(a=0, b=1)

#menampilkan hasil perhitungan kinerja dan luas AUC
print(paste("Luas AUC:", roc.auc@y.values))
```

Lampiran 6 Syntax R *Naïve Bayes Classifier* (Lanjutan)

```
#NBC TESTING
library(e1071)
library(caret)

#membaca data training & data testing
data.training = training_set10
data.test = test_set10

str(data.test)
str(data.training)

#membuat model
modell = naiveBayes(as.factor(klasifikasi)~., data.training)
modell

#melakukan prediksi
predict_result = predict(modell, data.test[, -19])
predict_result

#menghitung kinerja
performance.value = confusionMatrix(table(predict_result, data.test$klasifikasi), positive =
"Positif")
print(performance.value)

comparison_result = cbind(prediction = as.character(predict_result),
                           actual = as.character(data.test[, 19]))
comparison_result

write.csv(comparison_result, "cr_test10.csv", row.names = FALSE)

#menghitung kinerja
library(ROCR)
roc.prediction = prediction(as.numeric(as.factor(predict_result)),
                           as.numeric(as.factor(data.test[, 19])))
roc.tpr.fpr = performance(roc.prediction, "tpr", "fpr")
roc.auc = performance(roc.prediction, "auc")
plot(roc.tpr.fpr, col="red", lty=3)
abline(a=0, b=1)

#menampilkan hasil perhitungan kinerja dan luas AUC
print(paste("Luas AUC:", roc.auc@y.values))
```

Lampiran 7 Syntax *Python* Pencarian Parameter C dan Gamma Terbaik

```
import matplotlib.pyplot as plt
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import GridSearchCV
from sklearn.preprocessing import PowerTransformer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.feature_extraction.text import CountVectorizer
import re
import nltk.corpus
from sklearn.feature_extraction.text import TfidfVectorizer
import codecs
from sklearn import model_selection
from sklearn.datasets import make_classification
from sklearn.svm import SVC
from sklearn.metrics import confusion_matrix, classification_report, accuracy_score

from sklearn.metrics import precision_recall_curve, roc_auc_score
from sklearn.metrics import f1_score, roc_curve, auc

pip install pandas

from numpy import dtype
import pandas as pd
tfidf = pd.read_excel(r"data.xlsx")
kelas = pd.read_excel(r"data2.xlsx")
from sklearn.model_selection import train_test_split
x = tfidf
y = kelas['klasifikasi']
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2,
random_state = 0)

tuned_parameters = [{'kernel':['rbf'], 'C':[0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000], 'gamma': [0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000]}]
scores = ['recall']
for score in scores:
    print("# Tuning hyper-parameters for %s" % score)
    print()
    tuned = GridSearchCV(SVC(), tuned_parameters, scoring='%s_macro' % score)
    tuned.fit(x_train, y_train)
    print("Best parameters set found on development set:")
    print()
    print(tuned.best_params_)
    print()
    print("Grid scores on development set:")
    print()
    means = tuned.cv_results_['mean_test_score']
```

Lampiran 7 Syntax Python Pencarian Parameter C dan Gamma Terbaik (Lanjutan)

```
stds = tuned.cv_results_['std_test_score']
for mean, std, params in zip(means, stds, tuned.cv_results_['params']):
    print("%0.3f (+/-%0.03f) for %r"% (mean, std * 2, params))

import numpy as np
from sklearn.model_selection import StratifiedKFold
skf = StratifiedKFold(n_splits=10, random_state=0, shuffle=True)
skf.get_n_splits(x, y)
i = 1
for train_index, test_index in skf.split(x, y) :
    print("Fold ",i)
    print("TRAIN :",train_index,"TEST :",test_index)
    x_train ,x_test=x.loc[train_index] ,x.loc[test_index]
    y_train, y_test=y[train_index], y[test_index]
    i+=1
    modellin = SVC(kernel='rbf', probability=True, C=1000, gamma=1)
    clf_lin = modellin.fit(x_train, y_train)
    pred_labels_tr = modellin.predict(x_train)
    pred_labels_te = modellin.predict(x_test)
    cm_train = confusion_matrix(y_train, pred_labels_tr)
    cm_test = confusion_matrix(y_test, pred_labels_te)
    print('----- Evaluation on Training Data -----')
    score_tr = modellin.score(x_train, y_train)
    print(classification_report(y_train, pred_labels_tr))
    print('-----')
    print("Accuracy Train = {:.2f}%".format(score_tr*100))
    print("AUC Train: ", roc_auc_score(y_train, pred_labels_tr))
    print("CM Train = ")
    print(cm_train)
    print('----- Evaluation on Test Data -----')
    score_te = modellin.score(x_test, y_test)
    print(classification_report(y_test, pred_labels_te))
    print('-----')
    print("Accuracy Test = {:.2f}%".format(score_te*100))
    print("AUC Test: ", roc_auc_score(y_test,pred_labels_te))
    print("CM Test = ")
    print(cm_test)
    print('-----')
    print('b = ',clf_lin.intercept_)
    print('Indices of support vectors = ', clf_lin.support_)
    print('Number of support vectors for each class = ', clf_lin.n_support_)
    print('Coefficients of the support vector in the decision function = ',np.a
bs(clf_lin.dual_coef_))
    print('-----')
```

Lampiran 8 Syntax R Support Vector Machine

```
library(e1071)
library(caret)

#SVM TRAINING
data.train1 = training_setSVM1
data.train2 = training_setSVM1
#Model
svm <- svm(as.factor(klasifikasi)~., data.train1, kernel="radial", cost="1000", gamma="1")
#Prediksi
prediksi <- predict(svm, data.train2)
#menghitung kinerja
cm <- confusionMatrix(table(prediksi, data.train2$klasifikasi), positive = "Positif")
cm
#menghitung kinerja AUC
library(ROCR)
roc.prediction = prediction(as.numeric(as.factor(prediksi)),
                           as.numeric(as.factor(data.train2[,19])))
roc.tpr.fpr = performance(roc.prediction, "tpr", "fpr")
roc.auc = performance(roc.prediction, "auc")
plot(roc.tpr.fpr, col="red", lty=3)
abline(a=0, b=1)
#menampilkan hasil perhitungan kinerja dan luas AUC
print(paste("Luas AUC:", roc.auc@y.values))

#SVM TESTING
data.training = training_setSVM1
data.test = test_setSVM1
#Model
svm <- svm(as.factor(klasifikasi)~., data.training, kernel="radial", cost="1000", gamma="1")
#Prediksi
prediksi <- predict(svm, data.test)
#menghitung kinerja
cm <- confusionMatrix(table(prediksi, data.test$klasifikasi), positive = "Positif")
cm
#menghitung kinerja AUC
library(ROCR)
roc.prediction = prediction(as.numeric(as.factor(prediksi)),
                           as.numeric(as.factor(data.test[,19])))
roc.tpr.fpr = performance(roc.prediction, "tpr", "fpr")
roc.auc = performance(roc.prediction, "auc")
plot(roc.tpr.fpr, col="red", lty=3)
abline(a=0, b=1)
#menampilkan hasil perhitungan kinerja dan luas AUC
print(paste("Luas AUC:", roc.auc@y.values))
```

Lampiran 9 Syntax R *Word Cloud*

```
#wordcloud keseluruhan
wordcloud(text, min.freq = 4,max.words=100, random.order=F, colors=brewer.pal(8,"Dark2"))

#wordcloud sentimen

#Memanggil Sentimen Negatif
SentimenNegatif<-read.csv(file.choose(""), header = TRUE, stringsAsFactors = TRUE,
sep=";")

#wordcloud Sentimen Negatif
library(wordcloud)
teksnegatif = SentimenNegatif
corpustextnegatif <- Corpus(VectorSource(teksnegatif$text))
inspect(corpustextnegatif[1:100])
teks0 = corpustextnegatif
wordcloud(teks0, min.freq = 10,max.words=50, random.order=F,
colors=brewer.pal(8,"Dark2"))

#Memanggil Sentimen Positif
SentimenPositif<-read.csv(file.choose(""), header = TRUE, stringsAsFactors = TRUE, sep=";")

#wordcloud Sentimen positif
library(wordcloud)
tekspositif = SentimenPositif
corpustextpositif <- Corpus(VectorSource(tekspositif$text))
inspect(corpustextpositif[1:100])
teks1 = corpustextpositif
wordcloud(teks1, min.freq = 10,max.words=50, random.order=F,
colors=brewer.pal(8,"Dark2"))
```


Lampiran 10 Output R NBC

Iterasi	Data Training	Data Testing
1	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 429 4 Positif 1317 484</p> <p>Accuracy : 0.4087 95% CI : (0.3882, 0.4294) No Information Rate : 0.7816 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1206</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9918 Specificity : 0.2457 Pos Pred Value : 0.2687 Neg Pred Value : 0.9908 Prevalence : 0.2184 Detection Rate : 0.2167 Detection Prevalence : 0.8062 Balanced Accuracy : 0.6188</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.618753873021238"</p>	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 105 0 Positif 331 122</p> <p>Accuracy : 0.4068 95% CI : (0.3657, 0.4489) No Information Rate : 0.7814 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1218</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 1.0000 Specificity : 0.2408 Pos Pred Value : 0.2693 Neg Pred Value : 1.0000 Prevalence : 0.2186 Detection Rate : 0.2186 Detection Prevalence : 0.8118 Balanced Accuracy : 0.6204</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.620412844036697"</p>
2	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 558 3 Positif 1188 485</p> <p>Accuracy : 0.4669 95% CI : (0.446, 0.4878) No Information Rate : 0.7816 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1672</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9939 Specificity : 0.3196 Pos Pred Value : 0.2899 Neg Pred Value : 0.9947 Prevalence : 0.2184 Detection Rate : 0.2171 Detection Prevalence : 0.7489 Balanced Accuracy : 0.6567</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.656720043941186"</p>	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 135 1 Positif 301 121</p> <p>Accuracy : 0.4588 95% CI : (0.4169, 0.5011) No Information Rate : 0.7814 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1599</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9918 Specificity : 0.3096 Pos Pred Value : 0.2867 Neg Pred Value : 0.9926 Prevalence : 0.2186 Detection Rate : 0.2168 Detection Prevalence : 0.7563 Balanced Accuracy : 0.6507</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.65071815310573"</p>

Lampiran 10 Output R NBC (Lanjutan)

Iterasi	Data Training	Data Testing												
3	<p>Confusion Matrix and Statistics predict_result Negatif Positif</p> <table border="1"> <tr> <td>Negatif</td> <td>558</td> <td>4</td> </tr> <tr> <td>Positif</td> <td>1188</td> <td>484</td> </tr> </table> <p>Accuracy : 0.4664 95% CI : (0.4456, 0.4874) No Information Rate : 0.7816 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1662</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9918 Specificity : 0.3196 Pos Pred Value : 0.2895 Neg Pred Value : 0.9929 Prevalence : 0.2184 Detection Rate : 0.2167 Detection Prevalence : 0.7484 Balanced Accuracy : 0.6557</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.655695453777252"</p>	Negatif	558	4	Positif	1188	484	<p>Confusion Matrix and Statistics predict_result Negatif Positif</p> <table border="1"> <tr> <td>Negatif</td> <td>136</td> <td>0</td> </tr> <tr> <td>Positif</td> <td>300</td> <td>122</td> </tr> </table> <p>Accuracy : 0.4624 95% CI : (0.4204, 0.5047) No Information Rate : 0.7814 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1654</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 1.0000 Specificity : 0.3119 Pos Pred Value : 0.2891 Neg Pred Value : 1.0000 Prevalence : 0.2186 Detection Rate : 0.2186 Detection Prevalence : 0.7563 Balanced Accuracy : 0.6560</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.620412844036697"</p>	Negatif	136	0	Positif	300	122
Negatif	558	4												
Positif	1188	484												
Negatif	136	0												
Positif	300	122												
4	<p>Confusion Matrix and Statistics predict_result Negatif Positif</p> <table border="1"> <tr> <td>Negatif</td> <td>557</td> <td>6</td> </tr> <tr> <td>Positif</td> <td>1189</td> <td>482</td> </tr> </table> <p>Accuracy : 0.4651 95% CI : (0.4442, 0.486) No Information Rate : 0.7816 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1637</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9877 Specificity : 0.3190 Pos Pred Value : 0.2885 Neg Pred Value : 0.9893 Prevalence : 0.2184 Detection Rate : 0.2158 Detection Prevalence : 0.7480 Balanced Accuracy : 0.6534</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.653359904606313"</p>	Negatif	557	6	Positif	1189	482	<p>Confusion Matrix and Statistics predict_result Negatif Positif</p> <table border="1"> <tr> <td>Negatif</td> <td>142</td> <td>3</td> </tr> <tr> <td>Positif</td> <td>294</td> <td>119</td> </tr> </table> <p>Accuracy : 0.4677 95% CI : (0.4257, 0.5101) No Information Rate : 0.7814 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.162</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9754 Specificity : 0.3257 Pos Pred Value : 0.2881 Neg Pred Value : 0.9793 Prevalence : 0.2186 Detection Rate : 0.2133 Detection Prevalence : 0.7401 Balanced Accuracy : 0.6505</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.650548954730035"</p>	Negatif	142	3	Positif	294	119
Negatif	557	6												
Positif	1189	482												
Negatif	142	3												
Positif	294	119												

Lampiran 10 Output R NBC (Lanjutan)

Iterasi	Data <i>Training</i>	Data <i>Testing</i>
5	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 432 4 Positif 1314 484</p> <p>Accuracy : 0.41 95% CI : (0.3895, 0.4308) No Information Rate : 0.7816 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1216</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9918 Specificity : 0.2474 Pos Pred Value : 0.2692 Neg Pred Value : 0.9908 Prevalence : 0.2184 Detection Rate : 0.2167 Detection Prevalence : 0.8048 Balanced Accuracy : 0.6196</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.619612979550448"</p>	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 102 0 Positif 334 122</p> <p>Accuracy : 0.4014 95% CI : (0.3605, 0.4434) No Information Rate : 0.7814 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1178</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 1.0000 Specificity : 0.2339 Pos Pred Value : 0.2675 Neg Pred Value : 1.0000 Prevalence : 0.2186 Detection Rate : 0.2186 Detection Prevalence : 0.8172 Balanced Accuracy : 0.6170</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.61697247706422"</p>
6	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 574 4 Positif 1172 484</p> <p>Accuracy : 0.4736 95% CI : (0.4527, 0.4945) No Information Rate : 0.7816 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1721</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9918 Specificity : 0.3288 Pos Pred Value : 0.2923 Neg Pred Value : 0.9931 Prevalence : 0.2184 Detection Rate : 0.2167 Detection Prevalence : 0.7413 Balanced Accuracy : 0.6603</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.66027735526637"</p>	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 120 0 Positif 316 122</p> <p>Accuracy : 0.4337 95% CI : (0.3921, 0.476) No Information Rate : 0.7814 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1424</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 1.0000 Specificity : 0.2752 Pos Pred Value : 0.2785 Neg Pred Value : 1.0000 Prevalence : 0.2186 Detection Rate : 0.2186 Detection Prevalence : 0.7849 Balanced Accuracy : 0.6376</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.637614678899083"</p>

Lampiran 10 Output R NBC (Lanjutan)

Iterasi	Data Training	Data Testing
7	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 524 2 Positif 1222 486</p> <p>Accuracy : 0.4521 95% CI : (0.4313, 0.473) No Information Rate : 0.7816 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1557</p> <p>McNemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9959 Specificity : 0.3001 Pos Pred Value : 0.2845 Neg Pred Value : 0.9962 Prevalence : 0.2184 Detection Rate : 0.2175 Detection Prevalence : 0.7645 Balanced Accuracy : 0.6480</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.648008093440745"</p>	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 132 0 Positif 304 122</p> <p>Accuracy : 0.4552 95% CI : (0.4133, 0.4976) No Information Rate : 0.7814 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1596</p> <p>McNemar's Test P-Value : <2e-16</p> <p>Sensitivity : 1.0000 Specificity : 0.3028 Pos Pred Value : 0.2864 Neg Pred Value : 1.0000 Prevalence : 0.2186 Detection Rate : 0.2186 Detection Prevalence : 0.7634 Balanced Accuracy : 0.6514</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.651376146788991"</p>
8	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 549 3 Positif 1197 485</p> <p>Accuracy : 0.4628 95% CI : (0.442, 0.4838) No Information Rate : 0.7816 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1639</p> <p>McNemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9939 Specificity : 0.3144 Pos Pred Value : 0.2883 Neg Pred Value : 0.9946 Prevalence : 0.2184 Detection Rate : 0.2171 Detection Prevalence : 0.7529 Balanced Accuracy : 0.6541</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.654142724353558"</p>	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 144 1 Positif 292 121</p> <p>Accuracy : 0.4749 95% CI : (0.4328, 0.5173) No Information Rate : 0.7814 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1733</p> <p>McNemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9918 Specificity : 0.3303 Pos Pred Value : 0.2930 Neg Pred Value : 0.9931 Prevalence : 0.2186 Detection Rate : 0.2168 Detection Prevalence : 0.7401 Balanced Accuracy : 0.6610</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.661039254023161"</p>

Lampiran 10 Output R NBC (Lanjutan)

Iterasi	Data Training	Data Testing
9	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 437 2 Positif 1309 486</p> <p>Accuracy : 0.4132 95% CI : (0.3926, 0.4339) No Information Rate : 0.7816 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1253</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9959 Specificity : 0.2503 Pos Pred Value : 0.2708 Neg Pred Value : 0.9954 Prevalence : 0.2184 Detection Rate : 0.2175 Detection Prevalence : 0.8035 Balanced Accuracy : 0.6231</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.623094004093666"</p>	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 94 1 Positif 342 121</p> <p>Accuracy : 0.3853 95% CI : (0.3447, 0.4271) No Information Rate : 0.7814 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1034</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9918 Specificity : 0.2156 Pos Pred Value : 0.2613 Neg Pred Value : 0.9895 Prevalence : 0.2186 Detection Rate : 0.2168 Detection Prevalence : 0.8297 Balanced Accuracy : 0.6037</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.603699804481877"</p>
10	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 431 3 Positif 1315 485</p> <p>Accuracy : 0.41 95% CI : (0.3895, 0.4308) No Information Rate : 0.7816 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1223</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9939 Specificity : 0.2468 Pos Pred Value : 0.2694 Neg Pred Value : 0.9931 Prevalence : 0.2184 Detection Rate : 0.2171 Detection Prevalence : 0.8057 Balanced Accuracy : 0.6204</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.620351200871312"</p>	<p>Confusion Matrix and Statistics predict_result Negatif Positif Negatif 102 1 Positif 334 121</p> <p>Accuracy : 0.3996 95% CI : (0.3587, 0.4416) No Information Rate : 0.7814 P-Value [Acc > NIR] : 1</p> <p>Kappa : 0.1138</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.9918 Specificity : 0.2339 Pos Pred Value : 0.2659 Neg Pred Value : 0.9903 Prevalence : 0.2186 Detection Rate : 0.2168 Detection Prevalence : 0.8154 Balanced Accuracy : 0.6129</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.612874116408483"</p>

Lampiran 11 Output R SVM

Iterasi	Data Training	Data Testing
1	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 1628 336 Positif 118 152</p> <p>Accuracy : 0.7968 95% CI : (0.7795, 0.8133) No Information Rate : 0.7816 P-Value [Acc > NIR] : 0.04226</p> <p>Kappa : 0.2907</p> <p>Mcnemar's Test P-Value : < 2e-16</p> <p>Sensitivity : 0.31148 Specificity : 0.93242 Pos Pred Value : 0.56296 Neg Pred Value : 0.82892 Prevalence : 0.21844 Detection Rate : 0.06804 Detection Prevalence : 0.12086 Balanced Accuracy : 0.62195</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.621946181435788"</p>	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 398 82 Positif 38 40</p> <p>Accuracy : 0.7849 95% CI : (0.7485, 0.8183) No Information Rate : 0.7814 P-Value [Acc > NIR] : 0.4427</p> <p>Kappa : 0.2766</p> <p>Mcnemar's Test P-Value : 8.661e-05</p> <p>Sensitivity : 0.32787 Specificity : 0.91284 Pos Pred Value : 0.51282 Neg Pred Value : 0.82917 Prevalence : 0.21864 Detection Rate : 0.07168 Detection Prevalence : 0.13978 Balanced Accuracy : 0.62036</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.620356444578132"</p>
2	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 1625 341 Positif 121 147</p> <p>Accuracy : 0.7932 95% CI : (0.7758, 0.8098) No Information Rate : 0.7816 P-Value [Acc > NIR] : 0.09523</p> <p>Kappa : 0.2769</p> <p>Mcnemar's Test P-Value : < 2e-16</p> <p>Sensitivity : 0.3012 Specificity : 0.9307 Pos Pred Value : 0.5485 Neg Pred Value : 0.8266 Prevalence : 0.2184 Detection Rate : 0.0658 Detection Prevalence : 0.1200 Balanced Accuracy : 0.6160</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.615964124086906"</p>	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 405 80 Positif 31 42</p> <p>Accuracy : 0.8011 95% CI : (0.7655, 0.8334) No Information Rate : 0.7814 P-Value [Acc > NIR] : 0.1407</p> <p>Kappa : 0.3193</p> <p>Mcnemar's Test P-Value : 5.215e-06</p> <p>Sensitivity : 0.34426 Specificity : 0.92890 Pos Pred Value : 0.57534 Neg Pred Value : 0.83505 Prevalence : 0.21864 Detection Rate : 0.07527 Detection Prevalence : 0.13082 Balanced Accuracy : 0.63658</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.636580688825387"</p>

Lampiran 11 Output R SVM (Lanjutan)

Iterasi	Data Training	Data Testing
3	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 1624 336 Positif 122 152</p> <p>Accuracy : 0.795 95% CI : (0.7776, 0.8116) No Information Rate : 0.7816 P-Value [Acc > NIR] : 0.06464</p> <p>Kappa : 0.2869</p> <p>Mcnemar's Test P-Value : < 2e-16</p> <p>Sensitivity : 0.31148 Specificity : 0.93013 Pos Pred Value : 0.55474 Neg Pred Value : 0.82857 Prevalence : 0.21844 Detection Rate : 0.06804 Detection Prevalence : 0.12265 Balanced Accuracy : 0.62080</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.620800706063508"</p>	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 406 84 Positif 30 38</p> <p>Accuracy : 0.7957 95% CI : (0.7598, 0.8284) No Information Rate : 0.7814 P-Value [Acc > NIR] : 0.2224</p> <p>Kappa : 0.2887</p> <p>Mcnemar's Test P-Value : 6.909e-07</p> <p>Sensitivity : 0.3115 Specificity : 0.9312 Pos Pred Value : 0.5588 Neg Pred Value : 0.8286 Prevalence : 0.2186 Detection Rate : 0.0681 Detection Prevalence : 0.1219 Balanced Accuracy : 0.6213</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.621334035193262"</p>
4	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 1621 346 Positif 125 142</p> <p>Accuracy : 0.7892 95% CI : (0.7717, 0.8059) No Information Rate : 0.7816 P-Value [Acc > NIR] : 0.1995</p> <p>Kappa : 0.2622</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.29098 Specificity : 0.92841 Pos Pred Value : 0.53184 Neg Pred Value : 0.82410 Prevalence : 0.21844 Detection Rate : 0.06356 Detection Prevalence : 0.11952 Balanced Accuracy : 0.60970</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.609695697894954"</p>	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 414 86 Positif 22 36</p> <p>Accuracy : 0.8065 95% CI : (0.7712, 0.8384) No Information Rate : 0.7814 P-Value [Acc > NIR] : 0.08194</p> <p>Kappa : 0.3016</p> <p>Mcnemar's Test P-Value : 1.343e-09</p> <p>Sensitivity : 0.29508 Specificity : 0.94954 Pos Pred Value : 0.62069 Neg Pred Value : 0.82800 Prevalence : 0.21864 Detection Rate : 0.06452 Detection Prevalence : 0.10394 Balanced Accuracy : 0.62231</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.622311625808392"</p>

Lampiran 11 Output R SVM (Lanjutan)

Iterasi	Data Training	Data Testing
5	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 1632 335 Positif 114 153</p> <p>Accuracy : 0.799 95% CI : (0.7818, 0.8155) No Information Rate : 0.7816 P-Value [Acc > NIR] : 0.02352</p> <p>Kappa : 0.2966</p> <p>McNemar's Test P-Value : < 2e-16</p> <p>Sensitivity : 0.31352 Specificity : 0.93471 Pos Pred Value : 0.57303 Neg Pred Value : 0.82969 Prevalence : 0.21844 Detection Rate : 0.06849 Detection Prevalence : 0.11952 Balanced Accuracy : 0.62412</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.624116246972002"</p>	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 399 88 Positif 37 34</p> <p>Accuracy : 0.776 95% CI : (0.7391, 0.8099) No Information Rate : 0.7814 P-Value [Acc > NIR] : 0.6431</p> <p>Kappa : 0.2282</p> <p>McNemar's Test P-Value : 7.744e-06</p> <p>Sensitivity : 0.27869 Specificity : 0.91514 Pos Pred Value : 0.47887 Neg Pred Value : 0.81930 Prevalence : 0.21864 Detection Rate : 0.06093 Detection Prevalence : 0.12724 Balanced Accuracy : 0.59691</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.596913069634532"</p>
6	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 1633 330 Positif 113 158</p> <p>Accuracy : 0.8017 95% CI : (0.7846, 0.8181) No Information Rate : 0.7816 P-Value [Acc > NIR] : 0.01073</p> <p>Kappa : 0.3085</p> <p>McNemar's Test P-Value : < 2e-16</p> <p>Sensitivity : 0.32377 Specificity : 0.93528 Pos Pred Value : 0.58303 Neg Pred Value : 0.83189 Prevalence : 0.21844 Detection Rate : 0.07073 Detection Prevalence : 0.12131 Balanced Accuracy : 0.62953</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.629525566634744"</p>	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 401 94 Positif 35 28</p> <p>Accuracy : 0.7688 95% CI : (0.7316, 0.8032) No Information Rate : 0.7814 P-Value [Acc > NIR] : 0.78</p> <p>Kappa : 0.1807</p> <p>McNemar's Test P-Value : 3.28e-07</p> <p>Sensitivity : 0.22951 Specificity : 0.91972 Pos Pred Value : 0.44444 Neg Pred Value : 0.81010 Prevalence : 0.21864 Detection Rate : 0.05018 Detection Prevalence : 0.11290 Balanced Accuracy : 0.57462</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.57461648368175"</p>

Lampiran 11 Output R SVM (Lanjutan)

Iterasi	Data Training	Data Testing
7	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 1617 341 Positif 129 147</p> <p>Accuracy : 0.7896 95% CI : (0.7721, 0.8064) No Information Rate : 0.7816 P-Value [Acc > NIR] : 0.1854</p> <p>Kappa : 0.2695</p> <p>Mcnemar's Test P-Value : <2e-16</p> <p>Sensitivity : 0.3012 Specificity : 0.9261 Pos Pred Value : 0.5326 Neg Pred Value : 0.8258 Prevalence : 0.2184 Detection Rate : 0.0658 Detection Prevalence : 0.1235 Balanced Accuracy : 0.6137</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.613673173342347"</p>	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 412 81 Positif 24 41</p> <p>Accuracy : 0.8118 95% CI : (0.7769, 0.8434) No Information Rate : 0.7814 P-Value [Acc > NIR] : 0.04369</p> <p>Kappa : 0.3379</p> <p>Mcnemar's Test P-Value : 4.628e-08</p> <p>Sensitivity : 0.33607 Specificity : 0.94495 Pos Pred Value : 0.63077 Neg Pred Value : 0.83570 Prevalence : 0.21864 Detection Rate : 0.07348 Detection Prevalence : 0.11649 Balanced Accuracy : 0.64051</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.640509851105429"</p>
8	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 1626 337 Positif 120 151</p> <p>Accuracy : 0.7954 95% CI : (0.7781, 0.812) No Information Rate : 0.7816 P-Value [Acc > NIR] : 0.05833</p> <p>Kappa : 0.2866</p> <p>Mcnemar's Test P-Value : < 2e-16</p> <p>Sensitivity : 0.30943 Specificity : 0.93127 Pos Pred Value : 0.55720 Neg Pred Value : 0.82832 Prevalence : 0.21844 Detection Rate : 0.06759 Detection Prevalence : 0.12131 Balanced Accuracy : 0.62035</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.620348853585713"</p>	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 402 84 Positif 34 38</p> <p>Accuracy : 0.7885 95% CI : (0.7523, 0.8217) No Information Rate : 0.7814 P-Value [Acc > NIR] : 0.3632</p> <p>Kappa : 0.2739</p> <p>Mcnemar's Test P-Value : 6.458e-06</p> <p>Sensitivity : 0.3115 Specificity : 0.9220 Pos Pred Value : 0.5278 Neg Pred Value : 0.8272 Prevalence : 0.2186 Detection Rate : 0.0681 Detection Prevalence : 0.1290 Balanced Accuracy : 0.6167</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.616746879229959"</p>

Lampiran 11 Output R SVM (Lanjutan)

Iterasi	Data Training	Data Testing
9	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 1633 335 Positif 113 153</p> <p>Accuracy : 0.7995 95% CI : (0.7822, 0.8159) No Information Rate : 0.7816 P-Value [Acc > NIR] : 0.02077</p> <p>Kappa : 0.2976</p> <p>McNemar's Test P-Value : < 2e-16</p> <p>Sensitivity : 0.31352 Specificity : 0.93528 Pos Pred Value : 0.57519 Neg Pred Value : 0.82978 Prevalence : 0.21844 Detection Rate : 0.06849 Detection Prevalence : 0.11907 Balanced Accuracy : 0.62440</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.624402615815071"</p>	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 395 86 Positif 41 36</p> <p>Accuracy : 0.7724 95% CI : (0.7353, 0.8066) No Information Rate : 0.7814 P-Value [Acc > NIR] : 0.7156</p> <p>Kappa : 0.2318</p> <p>McNemar's Test P-Value : 9.447e-05</p> <p>Sensitivity : 0.29508 Specificity : 0.90596 Pos Pred Value : 0.46753 Neg Pred Value : 0.82121 Prevalence : 0.21864 Detection Rate : 0.06452 Detection Prevalence : 0.13799 Balanced Accuracy : 0.60052</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.600522634982704"</p>
10	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 1635 336 Positif 111 152</p> <p>Accuracy : 0.7999 95% CI : (0.7827, 0.8163) No Information Rate : 0.7816 P-Value [Acc > NIR] : 0.01829</p> <p>Kappa : 0.2973</p> <p>McNemar's Test P-Value : < 2e-16</p> <p>Sensitivity : 0.31148 Specificity : 0.93643 Pos Pred Value : 0.57795 Neg Pred Value : 0.82953 Prevalence : 0.21844 Detection Rate : 0.06804 Detection Prevalence : 0.11773 Balanced Accuracy : 0.62395</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.623950763337277"</p>	<p>Confusion Matrix and Statistics prediksi Negatif Positif Negatif 399 88 Positif 37 34</p> <p>Accuracy : 0.776 95% CI : (0.7391, 0.8099) No Information Rate : 0.7814 P-Value [Acc > NIR] : 0.6431</p> <p>Kappa : 0.2282</p> <p>McNemar's Test P-Value : 7.744e-06</p> <p>Sensitivity : 0.27869 Specificity : 0.91514 Pos Pred Value : 0.47887 Neg Pred Value : 0.81930 Prevalence : 0.21864 Detection Rate : 0.06093 Detection Prevalence : 0.12724 Balanced Accuracy : 0.59691</p> <p>'Positive' Class : Positif</p> <p>[1] "Luas AUC: 0.596913069634532"</p>

BIODATA PENULIS



Penulis bernama Nada Damai Bagaskara, merupakan kelahiran Kudus, Jawa Tengah dan dibesarkan di Surabaya, Jawa Timur. Pendidikan menengah yang ditempuh penulis diselesaikan di SMA Negeri 15 Surabaya, dimana selanjutnya melanjutkan ke Program Studi D-III Statistika Institut Teknologi Sepuluh Nopember (ITS) pada tahun 2016. Masa perkuliahan dilalui bersama angkatan 2016 Berdikari, yang mendapat gelar $\sigma 3$ dari Himpunan Mahasiswa Diploma Statistika ITS (HIMADATA-ITS). Pasca kelulusan dari Program D-III Statistika, penulis melatih diri dengan mengikuti banyak pelatihan diantaranya Bahasa Inggris, dan Digital Marketing. Dua tahun pasca kelulusan digunakan penulis untuk ikut berbagai pelatihan dan bekerja pada sebuah perusahaan industri di Sidoarjo, Jawa Timur selama satu

tahun tiga bulan. Pada pertengahan tahun 2021 penulis melanjutkan pendidikan ke jenjang Sarjana Terapan di Departemen Statistika Bisnis Institut Teknologi Sepuluh Nopember (ITS), dengan menempuh tiga semester. Penulis juga mengimplementasikan ilmu yang diperoleh saat kuliah melalui sebuah wadah bernama Datalogia di media sosial. Hobi yang dilakukan penulis saat waktu senggang adalah membaca buku atau artikel elektronik, mengikuti kajian dan tutorial, berdiskusi dengan orang lain, serta jalan-jalan menikmati keindahan alam. Masukan dan pesan dapat disampaikan kepada penulis melalui e-mail nadadamaibagaskara@gmail.com.