TUGAS AKHIR – SS 141501

TEXT MINING PADA AKUN RESMI PEMERINTAH KOTA SURABAYA DENGAN METODE REGRESI LOGISTIK, *SUPPORT VECTOR MACHINE* (SVM), DAN *NAÏVE BAYES CLASSIFIER* (NBC)

RAKHMAH WAHYU MAYASARI
NRP 062116 4500 0034

Dosen Pembimbing
Dr. Dra. Kartika Fithriasari, M.Si

PROGRAM STUDI SARJANA
DEPARTEMEN STATISTIKA
FAKULTAS MATEMATIKA, KOMPUTASI, DAN SAINS DATA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2018

TUGAS AKHIR – SS 141501

TEXT MINING PADA AKUN RESMI PEMERINTAH KOTA SURABAYA DENGAN METODE REGRESI LOGISTIK, *SUPPORT VECTOR MACHINE* (SVM), DAN *NAÏVE BAYES CLASSIFIER* (NBC)

RAKHMAH WAHYU MAYASARI
NRP 062116 4500 0034

Dosen Pembimbing
Dr. Dra. Kartika Fithriasari, M.Si

PROGRAM STUDI SARJANA
DEPARTEMEN STATISTIKA
FAKULTAS MATEMATIKA, KOMPUTASI, DAN SAINS DATA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2018

FINAL PROJECT – SS 141501

TEXT MINING OF SURABAYA GOVERNMENT OFFICIAL ACCOUNTS WITH LOGISTIC REGRESSION, SUPPORT VECTOR MACHINE (SVM), AND NAÏVE BAYES CLASSIFIER (NBC)

RAKHMAH WAHYU MAYASARI
SN 062116 4500 0034

Supervisor
Dr. Dra. Kartika Fithriasari, M.Si

UNDERGRADUATE PROGRAMME
DEPARTMENT OF STATISTICS
FACULTY OF MATHEMATICS, COMPUTING, AND DATA SCIENCE
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2018

LEMBAR PENGESAHAN

TEXT MINING PADA AKUN RESMI _PEMERINTAH KOTA SURABAYA DENGAN METODE_ REGRESI LOGISTIK, _SUPPORT VECTOR MACHINE_ (SVM), DAN _NAÏVE BAYES CLASSIFIER_ (NBC)

TUGAS AKHIR

Diajukan untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Sains
pada
Program Studi Sarjana Departemen Statistika
Fakultas Matematika, Komputasi, dan Sains Data
Institut Teknologi Sepuluh Nopember

Oleh :
RAKHMAH WAHYU MAYASARI
NRP. 062116 4500 0034

Disetujui oleh Pembimbing:
Dr. Dra. Kartika Fithriasari, M.Si ( )
NIP. 19691212 199303 2 002

Mengetahui,
Kepala Departemen

Dr. Suhartono
NIP. 19710929 199512 1 001

SURABAYA, JULI 2018

TEXT MINING PADA AKUN RESMI PEMERINTAH KOTA SURABAYA DENGAN METODE REGRESI LOGISTIK, _SUPPORT VECTOR MACHINE_ (SVM), DAN _NAÏVE BAYES CLASSIFIER_ (NBC)

Nama Mahasiswa : Rakhmah Wahyu Mayasari
NRP : 062116 4500 0034
Departemen : Statistika
Dosen Pembimbing : Dr. Dra. Kartika Fithriasari, M.Si

Abstrak

Pemerintah merupakan suatu lembaga administrasi yang berwenang atas kegiatan masyarakat dalam sebuah negara, kota dan sebagainya. Dalam menjalankan suatu kepemerintahan, Surabaya senantiasa terbuka dalam menerima sebuah masukan. Salah saru media penyampaiannya yaitu media sosial. Twitter merupakan salah satu media sosial yang telah ramai digunakan. Akun twitter resmi yang digunakan oleh pemerintah kota Surabaya adalah Sapawarga Surabaya (@SapawargaSby). Adapun akun radio Suara Surabaya (@e100ss) juga menjadi tempat berkeluh kesah masyarakat Surabaya. Data dari twitter diambil menggunakan sistem Application Programming Interface, dimana data kemudian dilakukan analisis sentimen. Berdasarkan hasil dari analisis diketahui bahwa metode terbaik yang digunakan adalah Support Vector Machine (SVM) kernel Radial Basis Function (RBF) karena memiliki ketepatan klasifikasi tertinggi dari pada metode SVM kernel Linear, Naïve Bayes Classifire (NBC), dan Regresi Logistik. Dilakukan pula analisis menggunakan Social Network Analysis (SNA) yang menghasilkan kesimpulan bahwa urutan tiga akun twitter yang sangat berpengaruh adalah @e100ss, @SapawargaSby, dan @BanggaSurabaya.

Kata Kunci : Analisis Sentimen, Naïve Bayes Classifier, Regresi Logistik, Social Network Analysis, Support Vector Machine.

(Halaman ini sengaja dikosongkan)

TEXT MINING OF SURABAYA GOVERNMENT OFFICIAL ACCOUNTS WITH LOGISTIC REGRESSION, SUPPORT VECTOR MACHINE (SVM), AND NAÏVE BAYES CLASSIFIER (NBC)

Student Name	**: Rakhmah Wahyu Mayasari**
Student Number	**: 062116 4500 0034**
Departement	**: Statistics**
Supervisor	**: Dr. Dra. Kartika Fithriasari, M.Si**

Abstract

Government is an administrative institution authorized for community activities in a country, city and so forth. In running a government, Surabaya is always open in receiving a suggestion. One of the media for delivering such suggestions is social media. Twitter is social media that has been used, wherever the official account used by Surabaya government is Sapawarga Surabaya (@SapawargaSby). The Radio Suara Surabaya official account (@e100ss) is also a place to express the opinion of Surabaya government performance. Data from twitter was taken using Application Programming Interface system, thereafter the data was analyzed sentiment. Based on the analysis results, it is known that the best method used is the Support Vector Machine (SVM) using Radial Basis Function kernel (RBF,) because it has the highest classification accuracy rather than kernel Linear, and also another method like Naïve Bayes Classifire (NBC), and Logistic Regression. In this study also perform an analysis using Social Network Analysis (SNA) which resulted in the conclusion that the sequence of three highly influential twitter account is @e100ss, @SapawargaSby, and @BanggaSurabaya.

Keyword: Analisis Sentimen, Naïve Bayes Classifier, Logistic Regression, Social Network Analysis, Support Vector Machine.

(Halaman ini sengaja dikosongkan)

KATA PENGANTAR

Puji syukur kehadirat Allah SWT yang telah memberikan rahmat, taufiq, dan hidayah-Nya sehingga penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul "***Text Mining pada Akun Resmi Pemerintah Kota Surabaya dengan Metode Regresi Logistik, Support Vector Machine (SVM), Dan Naïve Bayes Classifier (NBC)***". Penyusunan Tugas Akhir ini dapat terselesaikan dengan baik dan lancar karena tidak lepas dari dukungan berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Ibu Dr. Dra. Kartika Fithriasari, M.Si selaku dosen pembimbing yang telah membimbing, mengarahkan, dan memberikan dukungan bagi penulis untuk dapat menyelesaikan Tugas Akhir ini.
2. Bapak Dr. Suhartono selaku Ketua Jurusan Statistika ITS dan dosen wali yang telah memberikan nasehat, motivasi, serta bimbingan kepada penulis selama penulis menempuh pendidikan.dan menyediakan fasilitas untuk menyelesaikan Tugas Akhir
3. Bapak Dr. Sutikno, S.Si, M.Si selaku Ketua Program Studi Sarjana yang telah membimbing dan memotivasi penulis selama menjadi mahasiswa.
4. Ibu Dr. Irhamah, M.Si dan Ibu Pratnya Paramitha Oktaviana M.Si selaku dosen penguji yang telah memberikan saran-saran untuk kesempurnaan Tugas Akhir ini.
5. Seluruh dosen Jurusan Statistika ITS yang telah memberikan ilmu selama penulis menempuh pendidikan, beserta seluruh karyawan Jurusan Statistika ITS yang telah membantu kelancaran dan kemudahan dalam pelaksanaan kegiatan perkuliahan.
6. Bapak, Ibu, Kakak, dan semua keluarga di Tulungagung atas doa, kasih sayang, dukungan, semangat dan segalanya yang telah diberikan untuk penulis sehingga dilancarkan dalam menyelesaikan Tugas Akhir ini.

7. Fransiska Kristin D., Mary Happy, Idris Hibbatullah A.K., Adimas Raka D., Pramavidha Cory, Renita Elizabeth S., Winindyah Ayu L., Nur Lailatul F., Auliya Azizah, Hanum Kumala, Guruh Arya S., Danuja Wijayanto, Dimas Tiar, Safira Nur A., Egan Pradana, Cristian Febrianto, Peter Andreas, Syafira Khayam, Vincentius Aldi M., Prasetiyo Lumadi., Firsty Swastika, Frans Loekito, Hans Kristian, Ardianto Tjahjono, Arum Adiwida H., Stefanie Enrica, Estka, Hemas Mutia A., Hera Monica, Nabibah Hanun, Erik Noer M., Firdausy Azmi R., dan Aziz Ainun Najib yang telah membantu ketika penulis memiliki masalah akademik maupun non akademik dengan memberikan semangat, perhatian dan waktu selama menjalani hari-hari di masa perkuliahan.
8. Inung Anggun S, Zuyyin Inesa, Lely Presty, Camelia Nanda, Yongky Choirul A., Raras Anasi, Siti Azizah, dan Rima Kusumawati yang selalu memberi dukungan, semangat dan hiburan saat bertukar cerita baik susah maupun duka selama kuliah.
9. Teman-teman S1 LJ Jurusan Statistika ITS Angkatan 2016 dan Keluarga Legendary Σ24 yang telah bekerja sama dengan baik selama penulis menempuh pendidikan, serta memberikan kenangan yang berharga bagi penulis.
10. Semua pihak yang telah memberikan dukungan yang tidak dapat disebutkan satu persatu oleh penulis.

Penulis menyadari bahwa laporan Tugas Akhir ini masih jauh dari kata sempurna, oleh karena itu penulis sangat mengharapkan kritik dan saran yang membangun agar berguna untuk perbaikan berikutnya. Semoga laporan Tugas Akhir ini bermanfaat.

Surabaya, Juli 2018

Penulis

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

(Halaman ini sengaja dikosongkan)

DAFTAR LAMPIRAN

(Halaman ini sengaja dikosongkan)

BAB I
PENDAHULUAN

1.1 Latar Belakang

Pemerintah merupakan suatu lembaga administrasi yang berwenang atas kegiatan masyarakat dalam sebuah negara, kota dan sebagainya. Sebagaimana pada Undang-Undang Nomor 32 Tahun 2004 bahwa suatu pemerintah daerah berwenang untuk mengatur urusan pemerintahannya menurut asas otonomi daerah, dimana pemerintah daerah ini meliputi gubernur, bupati atau walikota, serta perangkat daerah sebagai unsur penyelenggara pemerintah daerah. Pemerintah kota Surabaya merupakan salah satu pemerintahan yang melaksanakan asas otonom (Mariati, 2012). Pelaksanaan asas otonom oleh pemerintah Surabaya ini menghasilkan berbagai prestasi dengan banyaknya penghargaan yang telah diperoleh. Prestasi Pemerintah Kota (Pemkot) Surabaya dalam pembangunan kota dan pembangunan manusia kerap mendapatkan apresiasi dari berbagai pihak salah satunya adalah penghargaan dari Samkarya Parasamya Purnakarya Nugraha. Samkarya Parasamya Purnakarya Nugraha adalah sebuah tanda penghargaan yang diberikan kepada pemerintah daerah yang menunjukkan hasil karya tertinggi pelaksanaan pembangunan selama lima tahun. Berbagai keberhasilan yang dapat dilihat dari peningkatan indeks pembangunan manusia di Surabaya ini tidak luput dari program peningkatan sumber daya manusia. Selain peningkatan sumber daya manusia juga dilakukan upaya peningkatan dalam pengelolaan lingkungan sehingga Surabaya dapat memperoleh piala adipura kencana secara berturut-turut sejak tahun 2010 hingga 2017 (Ardianto, 2017).

Terdapat beberapa bagian dalam pemerintahan Surabaya, salah satunya adalah humas (hubungan masyarakat). Salah satu tugas humas pemerintah adalah menyebarluaskan informasi dan kebijakan pemerintah sesuai dengan institusi/lembaga masing-masing kepada publik, menampang dan mengelola aspirasi

masyarakat, serta membangun kepercayaan publik guna menjaga citra dan reputasi pemerintah. Penyebaran informasi oleh humas pemerintah dapat dikomunikasikan melalui media tradisional, media konvensional, dan media baru. Komunikasi mengunakan media baru atau teknologi internet telah terbukti dapat menjangkau langsung dengan cepat kepada semua pihak. Hal ini didukung dengan jumlah pengguna internet di Indonesia pada tahun 2015 mencapai 132,7 juta atau setara dengan 51,7% dari total penduduk 256,2 juta jiwa, dimana angka ini melesat 16,8% dari tahun 2014 (Tim APJII, 2016).

Banyaknya pengguna internet ini membuat masyarakat lebih kritis dalam menanggapi kondisi sekitar. Kinerja pemerintahpun tak luput dari sorotan masyarakat untuk mampu memenuhi kebutuhan-kebutuhan masyarakat dalam berbagai aspek. Tuntutan kebutuhan ini yang kelak akan membangun pemerintah menjadi lebih baik lagi, sehingga adanya pendapat mengenai performa pemerintah merupakan hal yang sangat dibutuhkan demi perbaikan pemerintahan. Adapun pendapat yang diberikan kepada pemerintah ini yang bersifat positif dan juga negatif. Pendapat positif ini meliputi apresiasi masyarakat mengenai hasil kinerja yang memuaskan oleh pemerintah kota Surabaya, sedangkan pendapat negatif banyak berasal dari keluhan-keluhan terhadap pelayanan yang dirasa belum optimal. Salah satu media yang dapat menampung pendapat masyarakat ini adalah sosial media yang mana 40% dari total pengguna internet adalah penggila sosial media. Salah satu media sosial yang digunakan oleh pemerintah kota Surabaya adalah twitter. Twitter merupakan salah satu layanan jejaring sosial yang memiliki tampilan mudah digunakan dengan maksimal 140 karakter yang biasa disebut dengan *tweet*. Akun twitter yang digunakan oleh pemerintah kota Surabaya adalah Sapawarga Surabaya (@SapawargaSby). Pada akun tersebut banyak pendapat dari masyarakat mengenai keadaan sekitar dan apresiasi kinerja pemerintahan. Selain pada akun twitter Sapawarga Surabaya adapun akun radio Suara Surabaya (@e100ss) juga menjadi tempat berkeluh kesah

masyarakat Surabaya. Komentar atau Pendapat dari masyarakat ini merupakan data penting yang dapat digunakan sebagai bahan evaluasi terhadap performa pemerintah kota Surabaya sehingga dapat lebih baik untuk kedepannya. Pengklasifikasian pendapat menjadi positif atau negatif selama ini dilakukan dengan cara manual dengan membaca satu persatu dari *tweet* yang diperuntukkan pemerintah kota Surabaya. Oleh karena itu peneliti merasa memiliki metode yang lebih tepat digunakan untuk pengklasifikasian pendapat ini dengan metode statistika.

Metode statistika yang dapat digunakan untuk mengklasifikasi data sangatlah banyak, akan tetapi pengklasifikasian ini mengerucut pada data teks dimana metode yang tepat digunakan adalah *text mining*. *Text mining* adalah satu cabang dari ilmu data *mining* yang menganalisis suatu data berupa teks. *Text mining* dapat digunakan untuk beberapa proses diantaranya penemuan *rule* baru dengan algoritma pengelompokan, asosiasi, dan *ranking*. Dari ketiga fungsi tersebut, *text mining* paling banyak dilakukan pada proses pengelompokan. Terdapat dua jenis metode pengelompokan teks, yaitu *text clustering* dan *text classification*. Perbedaan antara *text clustering* dan *text classification* ada pada penentuan kelompok yang dibentuk. *Text clustering* merupakan proses menemukan sebuah struktur kelompok yang belum terlihat dari sekumpulan dokumen. Sedangkan *text classification* merupakan proses untuk membentuk golongan dari dokumen berdasarkan pada kelas kelompok yang sudah diketahui sebelumnya. Berdasarkan pengertian ini, dapat dinyatakan bahwa proses klasifikasi merupakan proses yang tepat dan lebih mudah untuk dilakukan *monitoring*, karena terdapat target kelas yang akan dituju dalam analisisnya. Terdapat beberapa metode yang dapat digunakan untuk *text clustering* ini, diantaranya adalah *K-means Algoritm, Support Vector Machine* (SVM), *EM algoritm, PageRank, AdaBoost, K-nearest neighbor classification, Naïve Bayes Classifier* (NBC), dan *Classification and Regression Trees* (CART). Dari berbagai metode yang dapat digunakan, penelitian

ini akan menggunakan metode *Naïve Bayes Classifier* (NBC) dan *Support Vector Machine* (SVM). Metode NBC adalah metode yang paling sering digunakan dalam kasus *text classification*, hal ini dikarenakan NBC menggunakan algoritma yang sederhana namun memiliki hasil akurasi yang baik. Hampir sama dengan NBC, SVM merupakan metode yang cepat dan efektif pada pengklasifikasian data. Adapun metode klasik yang akan digunakan untuk membandingkan ketepatan klasifikasi dengan metode-metode yang telah disebutkan yaitu Regresi Logistik Biner. Regresi Logistik Biner merupakan regresi yang bertujuan untuk melihat pola hubungan antara variabel prediktor dan variabel respon dimana variabel responnya berupa data biner yaitu 1 sebagai sukses dan 0 sebagai gagal, dalam kasus ini 1 sebagai pendapat positif dan 0 sebagai pendapat negatif. Analisis *Social Network Analysis* (SNA) juga dilakukan pada penelitian ini, untuk identifikasi pengguna twitter yang berpengaruh pada pendapat masyarakat mengenai pemerintah kota Surabaya.

Penelitian yang pernah dilakukan mengenai analisis klasifikasi pada tahun 2015 dengan judul Klasifikasi Berita Indonesia Menggunakan Metode *Naïve Bayesian Classification* (NBC) dan *Support Vector Machine* (SVM) dengan *Confix Stripping Stemmer* oleh Dio Ariadi menghasilkan kesimpulan bahwa SVM *kernel* linier dan *kernel* RBF menghasilkan ketepatan klasifikasi yang sama dan bila dibandingkan dengan NBC maka SVM lebih baik. Pada tahun yang sama juga dilakukan penelitian oleh Riska Prakasita Sahitayakti yang mengangkat topik klasifikasi kesejahteraan rumah tangga di provinsi Papua menggunakan metode Regresi Logistik dan *Support Vector Machine* dengan hasil ketepatan klasifikasi menggunakan metode SVM lebih baik dari pada menggunakan metode regresi logistik biner. Penelitian lain dilakukan oleh Nuke Yulnida Aden F. pada tahun 2016 yang membahas mengenai analisis komentar di Sosial Media Twitter @SapawargaSby dan @e100ss menjadi kategori positif, netral, dan negatif menggunakan metode algoritma *Naïve Bayes* dan *Support Vector*

Machine. Hasil yang diperoleh adalah nilai akurasi dengan menggunakan metode *Support Vector Machine* lebih besar dari nilai akurasi menggunakan metode *Naïve Bayes*. Penelitian tentang klasifikasi berita online juga pernah dilakukan oleh Siti Nur Asiyah dengan metode *Support Vector Machine* dan *K-Nearest Neighbor* (KNN) dengan kesimpulan bahwa SVM lebih baik daripada KNN, penelitian ini dilakukan pada tahun 2016. Penelitian mengenai *Naïve Bayes Classifier* dan Regresi Logistik telah dilakukan pada tahun 2017 dengan judul Pengoptimalan *Naïve Bayes* dan Regresi Logistik menggunakan Algoritma Genetika untuk Data Klasifikasi oleh Abdurrahman Salim dengan hasil ketepatan klasifikasi mengunakan *Naïve Bayes* lebih tinggi dari metode Regresi Logistik Biner.

Pada penelitian ini variabel yang digunakan terdiri dari variabel independen yaitu kata dasar *tweet* yang telah dilakukan praproses data dan variabel dependen yaitu klasifikasi sentimen *tweet* positif atau negatif. Tujuan dari dilakukan penelitian ini adalah melakukan analisis sentimen mengenai tanggapan masyarakat terhadap pemerintah kota Surabaya berdasarkan media sosial twitter. Metode yang digunakan adalah *Naïve Bayes Classifier, Support Vector Machine* dan Regresi Logistik Biner sebagai perbandingan metode klasiknya. Dengan adanya penelitian ini diharapkan dapat membantu humas pemerintah kota Surabaya dalam melakukan analisis terkait pendapat masyarakat terhadap pemerintahan kota Surabaya.

1.2 Perumusan Masalah

Pengklasifikasian pendapat masyarakat Surabaya sekarang ini masih menggunakan cara manual, sehingga perlu dilakukan upaya untuk mempercepat pengklasifikasian menggunakan metode statistika. Metode klasifikasi yang tepat adalah *Naïve Bayes Classifier*, *Support Vector Machine*, dan Regresi Logistik Biner. Dari ketiga metode tersebut akan dipilih satu metode terbaik yang kemudian akan dalam mengklasifikasikan pendapat masyarakat Surabaya. Dilakukan pula analisis untuk mengetahui

pengguna twitter yang berpengaruh pada pendapat masyarakat Surabaya dengan metode *Social Network Analysis.*

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah dipaparkan sebelumnya, tujuan yang ingin dicapai dalam penelitian ini adalah sebagai berikut.

1. Mengetahui karakteristik data *tweet* masyarakat terhadap pemerintah kota Surabaya.
2. Mengetahui praproses teks pada data *tweet* terhadap pemerintah kota Surabaya.
3. Mengetahui kata yang sering muncul dengan menggunakan visualisasi *Word Cloud*.
4. Mengetahui perbandingan performa klasifikasi antara metode *Naïve Bayes Classifier*, *Support Vector Machine* dan Regresi Logistik Biner.
5. Mengetahui hasil identifikasi pengguna twitter yang berpengaruh pada pendapat masyarakat mengenai performa pemerintah kota Surabaya menggunakan metode *Social Network Analysis*.

1.4 Manfaat Penelitian

Hasil penelitian ini diharapkan dapat bermanfaat untuk beberapa aspek sebagai berikut.

1. Membantu humas pemerintah kota Surabaya dalam pengolahan data teks dari *tweet* terhadap pemerintah kota Surabaya menjadi lebih cepat.
2. Memberikan tambahan informasi kepada pemerintah kota Surabaya terhadap pendapat dari masyarakan melalui *tweet* terhadap pemerintah kota Surabaya.

1.5 Batasan Penelitian

Batasan masalah yang digunakan pada penelitian ini adalah sebagai beirkut.

1. Penelitian hanya menggunakan akun twitter pemerintah kota surabya (@SapawargaSby) dan Radio Suara Surabaya (@e100ss)
2. Penelitian menganggap semua pemilik akun yang melakukan *tweet* terhadap pemerintah kota Surabaya adalah sama
3. Data yang digunakan adalah data *tweet* terhadap pemerintah kota Surabaya pada 28 September 2017 sampai dengan 7 Mei 2018.

(Halaman ini sengaja dikosongkan)

BAB II
TINJAUAN PUSTAKA

2.1 *Text Mining*

Text mining merupakan salah satu cabang ilmu *data mining* yang menganalisis suatu data berupa *text*. *Text mining* adalah suatu langkah analisis teks yang dilakukan otomatis oleh komputer untuk menggali informasi yang berkualitas dari suatu rangkaian teks yang terangkum dalam sebuah dokumen. Ide awal pembuatan *text mining* adalah untuk menemukan pola-pola informasi yang dapat digali dari teks yang tidak terstruktur (Weiss, Indurkhya, Zhang, & Damerau, 2005). Dengan demikian, *text mining* mengacu juga kepada istilah *text data mining* atau penemuan pengetahuan dari basis data teks. Saat ini, *text mining* telah mendapatkan perhatian dalam berbagai bidang, antara lain dibidang keamanan, biomedis, pengembangan perangkat lunak dan aplikasi, media *online*, pemasaran, dan akademik. Seperti halnya dalam *data mining*, aplikasi *text mining* pada suatu studi kasus, harus dilakukan sesuai prosedur analisis. Langkah awal sebelum suatu data teks dianalisis menggunakan metode-metode dalam *text mining* adalah melakukan *pre-processing* teks, sehingga setelah didapatkan data yang siap diolah, analisis *text mining* dapat dilakukan.

Text mining dapat digunakan untuk proses penemuan *rule* baru dengan algoritma pengelompokan, asosiasi, dan *ranking*. Beberapa fungsi tersebut, yang paling banyak dilakukan adalah proses pengelompokan. Terdapat dua jenis metode pengelompokan teks, yaitu *text clustering* dan *text classification*. *Text clustering* berhubungan dengan proses menemukan sebuah struktur kelompok yang belum terlihat (tidak terpandu atau *unsupervised*) dari sekumpulan dokumen. Sedangkan, *text classification* dapat dianggap proses untuk membentuk golongan (kelas-kelas) dari dokumen berdasarkan pada kelas kelompok yang sudah diketahui sebelumnya (terpandu atau *supervised*). Berdasarkan pengertian ini, dapat dinyatakan bahwa proses

klasifikasi (*supervised*) merupakan proses yang lebih mudah dilakukan *monitoring*, karena terdapat target kelas yang akan dituju dalam analisisnya. Beberapa contoh metode yang dapat digunakan untuk klasifikasi suatu data teks adalah dengan *Naïve Bayes Classifier* (NBC) dan *Support vectorMachine* (SVM).

Salah satu lanjutan dari *text classification* adalah analisis sentimen, dimana analisis sentimen merupakan suatu analisis komputasi yang meneliti tentang pendapat, perasaan, dan emosi yang diungkapkan dalam sebuah tulisan. Tulisan ini seringkali bersifat subjektif mengenai kejadian dan aktifitas disekitar. Fokus utama dari analisis sentimen adalah mengelompokan suatu pendapat menjadi positif atau negatif, dimana pendapat tersebut dapat berupa kalimat ataupun dokumen. Analisis sentimen juga dapat menyatakan perasaan emosional seperti gembira, marah, atau sedih. Ekspresi ini dapat mengacu pada fokus topik tertentu, yang mana pernyataan pada satu topik dapat bermakna berbeda dengan pernyataan yang sama pada subjek yang berbeda. Oleh karena itu perlu dilakukan penentuan elemen-elemen dari sebuah produk yang sedang dibicarakan sebelum melakukan analisis sentimen (Bing, 2010).

2.2 Praproses Teks

Praproses teks merupakan tahapan-tahapan yang dilakukan sebelum mengolah data yang telah diperoleh. Praproses ini perlu dilakukan karena data mentah yang diperoleh merupakan data yang belum terstruktur dan belum dapat dilakukan proses *text mining.* Adapun tahapan-tahapan praproses data adalah sebagai berikut.

1. *Cleansing*, tahap *cleansing* merupakan tahapan untuk menghilangkan kata yang tidak diperlukan misalnya karakter HTML, *link URL, username (@username), emoticons*, dan *hastag (#).* Tahap *cleansing* ini diperlukan karena kata-kata tersebut dianggap *noise* yang tidak diperlukan pada proses data (Buntoro, Adji, & Purnamasari, 2014).

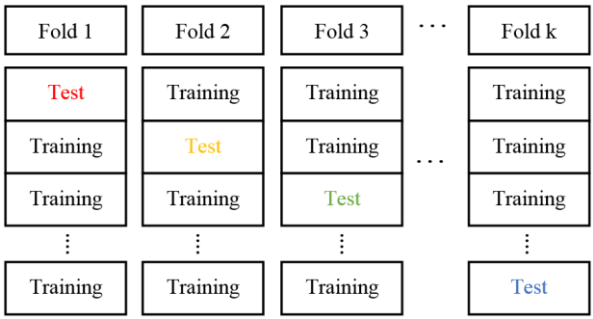
2. *Case Folding*, tahap *case folding* merupakan tahapan untuk menghilangkan angka dan tanda baca, serta mengubah karakter teks menjadi huruf kecil semua. Sistem kerja *case folding* yaitu memproses huruf alphabet "a" sampai dengan "z", sehingga karakter diluar alphabet akan dihilangkan seperti halnya tanda baca dan angka (Weiss, Indurkhya, Zhang, & Damerau, 2005).
3. *Stemming*, tahap *stemming* ini merupakan tahap mendapatkan kata dasar. Sistem kerja tahap *stemming* ini adalah menghilangkan awalan, akhiran, sisipan, dan *confixes* (kombinasi dari awalan dan akhiran) (Ariadi & Fithriasari, 2015).
4. *Stopwords*, tahap *stopwords* merupakan tahap penghilangan kosakata yang bukan termasuk kata unik atau tidak menyampaikan pesan apapun secara signifikan pada teks. Kosakata yang dimaksud seperti kata penghubung dan kata keterangan misalnya "oleh", "di", "yang", "jadi", dan sebagainya (Dragut, Fang, Sistla, & Yu, 2009).
5. *Tokenizing*, tahap *tokenizing* merupakan tahapan memutuskan kata per kata pada kalimat. Tahapan ini bertujuan untuk memecah yang semula berupa kalimat menjadi potongan-potongan kata, sehingga urutan *string* akan terputus menjadi potongan-potongan kata penyusunnya (Bing, 2010).
6. *Term Frequency-Inverse Document Frequency* (TF-IDF) yang digunakan untuk mengubah *text* menjadi variabel dengan rumus sebagai berikut.

$$w_{ij} = tf_{ij} \times idf \text{ dengan } idf = \log\left(\frac{N}{df_j}\right) \qquad (2.1)$$

dimana w_{ij} adalah bobot dari kata *i* pada artikel ke-*j*, *N* merupakan jumlah seluruh dokumen tf_{ij} adalah jumlah kemunculan kata *i* pada dokumen *j*, df_{ij} adalah jumlah artikel *j* yang mengandung kata *i* (Ariadi & Fithriasari, 2015).

2.3 K-*fold Cross validation*

Salah satu metode yang digunakan untuk mempartisi data menjadi data *training* dan data *testing* adalah *K-fold cross validation*. Pada metode ini secara berulang-ulang dilakukan pembagian data menjadi data *training* dan data *testing*, dimana setiap data mendapat kesempatan menjadi data *testing*. *K-fold cross validation* banyak digunakan karena dapat mengurangi bias yang terjadi dalam pengambilan sampel (Gokgoz & Subasi, 2015). *K* disini adalah angka partisi data yang digunakan untuk pembagian *training-testing* (Gokgoz & Subasi, 2015). Ilustrasi pembagian data *training-testing* menggunakan *K-fold cross validation* terdapat pada Gambar 2.1 berikut ini.



Gambar 2.1 Ilustrasi Pembagian Data

2.4 *Naïve Bayes Classifier* (NBC)

Teorema bayes merupakan teorema yang mengacu pada konsep probabilitas bersyarat yang secara umum dinotasikan sebagaimana persamaan 2.2 (Siang, 2005). Berdasarkan pada persamaan 2.2 diketahui $P(A|B)$ memiliki arti bahwa peluang kejadian A apabila B terjadi. Sedangkan *Naïve Bayes Classifier* (NBC) merupakan metode yang digunakan untuk mencari nilai probabilitas tertinggi untuk mengklasifikasi data uji pada kategori yang paling (Feldman & James, 2007). Tujuan utama dilakukan metode ini untuk mengklasifikasikan teks, dimana metode NBC

memiliki algoritma yang sederhana tetapi akurasinya tinggi. Terdapat dua tahap dalam klasifikasi *tweet*. Tahap pertama adalah pelatihan (*training*) terhadap *tweet* yang telah diketahui kategorinya. Sedangkan tahap kedua adalah proses klasifikasi (*testing*) *tweet* yang belum diketahui kategorinya. Pada algoritma NBC setiap dokumen direpresentasikan dengan pasangan atribut $a_1, a_2, ..., a_n$ dimana a_1 adalah kata pertama, a_2 adalah kata kedua, dan seterusnya. Sedangkan *V* adalah himpunan kategori *tweet*. Pada saat klasifikasi algoritma akan mencari probabilitas tertinggi dari semua kategori dokumen yang diujikan (V_{MAP}) sebagaimana persamaan 2.3. Adapun nilai $P(v_j)$ yang dihitung pada saat *training* dan nilai $P(a_i \mid v_j)$ untuk probabilitas kata a_i untuk setiap kategori dapat diperoleh dari persamaan 2.4 dan 2.5

$$P(A \mid B) = \frac{P(A)P(B \mid A)}{P(B)} \tag{2.2}$$

$$V_{MAP} = \arg\max_{v_j \in V} P(v_j) \prod_i^n P(a_i \mid v_j) \tag{2.3}$$

$$P(V_j) = \frac{|\, doc\ j \,|}{|\, training \,|} \tag{2.4}$$

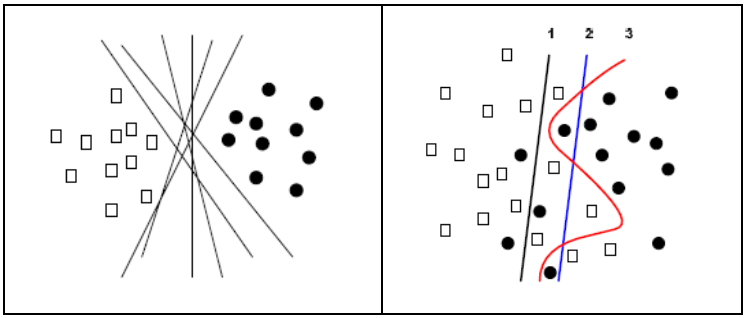
$$P(a_i \mid v_j) = \frac{n_i + 1}{|n + kosa\, kata|} \tag{2.5}$$

dimana *i* = 1, 2, 3, …, jumlah kata kunci dan *j* = 1, 2, 3, sampai dengan jumlah *tweet* yang digunakan. |*doc j*| merupakan jumlah *tweet* yang memiliki kategori *j* dalam *training*. Sedangkan, |*training*| merupakan jumlah *tweet* dalam contoh yang digunakan untuk *training*. n_i adalah jumlah kemunculan kata a_i dalam *tweet* yang berkategori v_j , sedangkan *n* adalah banyaknya

seluruh kata dalam *tweet* dengan kategori v_j dan |*kosa kata*| adalah banyaknya kata dalam data *training*.

2.5 *Support Vector Machine* (SVM)

Support Vector Machine (SVM) pertama kali diperkenalkan pada akhir 1979 oleh Vapnik yang selanjutnya mendapatkan perhatian yang lebih pada tahun 1992 bersama dengan Boser (Hardle, Prastyo, & Hafner, 2014). SVM merupakan metode machine learning yang dapat digunakan untuk prediksi dan klasifikasi. Konsep dasar SVM sebenarnya merupakan kombinasi dari beberapa konsep yang pernah ada sebelumnya dalam mengatasi permasalah terutama pada kasus klasifikasi dan prediksi. Konsep klasifikasi dengan menggunakan metode SVM adalah mencari persamaan *hyperplane* terbaik antar dua kelas klasifikasi. Persamaan *hyperplane* dikatakan baik jika memiliki margin terbesar. Margin adalah dua kali jarak antara *hyperplane* dan *support vector*, dimana *support vector* adalah titik yang berada paling dekat dengan *hyperplane.* Pada dasarnya SVM dikembangkan dengan prinsip *linier classifier* yang dapat dibedakan menjadi dua yaitu *Linear separable* dan non*separable* sebagaimana pada Gambar 2.2 (Nugroho, Witarto, & Handoko, 2003).

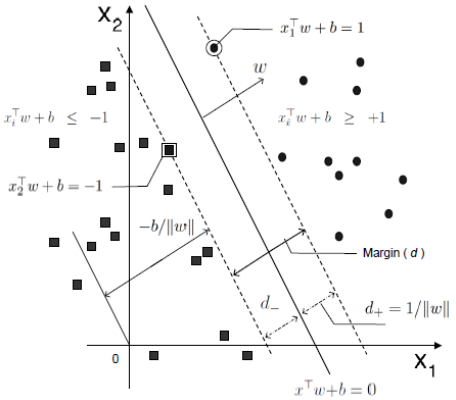


Gambar 2.2 Ilustrasi *Linear*y *Separable* (kiri) *Linear*y Non*separable* (Kanan)

(Sumber : Hardle, Prastyo, & Hafner, 2014)

2.5.1 *Support Vector Machine Linier Separable*

Setiap observasi terdiri dari sepasang *p* prediktor $\mathbf{x}_i = \left(x_{i1}, x_{i2}, \ldots, x_{ip}\right) \in \mathbb{R}^p$ dimana *i* berjalan dari 1 sampai dengan n. n adalah banyak data dan label kelas dari data $\mathbf{x}_i$ yang dinotasikan $y_i \in \mathbf{y} = \{-1,1\}$. Apabila $\mathbf{x}_i$ merupakan anggota dari kelas (+1) maka $\mathbf{x}_i$ diberi label $y_i = +1$, sebaliknya jika $\mathbf{x}_i$ merupakan anggota dari kelas (-1) maka $\mathbf{x}_i$ diberi label $y_i = -1$. sehingga diperoleh pasangan $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)$. Pasangan tersebut merupakan himpunan data *training* dari kelas yang akan diklasifikasikan menggunakan metode SVM. Untuk mendapatkan gambaran mengenai klasifikasi *Linear*ly *separable*, sebagaimana pada Gambar 2.3 akan disajikan ilustrasi metose SVM pada kasus *Linear*y *separable*.



Gambar 2.2 Ilustrasi *Linear*y *Separable* Klasifikasi SVM

(Sumber : Hardle, Prastyo, & Hafner, 2014)

Konsep utama dari SVM pada kasus *Linear separable* adalah menetapkan pemisah *Linear* antar dua *vector* yang ditetapkan sebagai $\mathbf{x}'\mathbf{w} = \sum_{i=1}^{n} x_i w_i$. Bidang pemisah atau yang

disebut juga dengan *hyperplane* dituliskan sebagaimana pada persamaan 2.6 berikut.

$$f(x) = \mathbf{x}'\mathbf{w} + b = 0 \tag{2.6}$$

dimana:

Bidang pembatas kelas pertama : $\mathbf{x}_i'\mathbf{w} + b \geq +1$ untuk $\mathbf{y}_i = +1$

Bidang pembatas kelas kedua : $\mathbf{x}_i'\mathbf{w} + b \leq -1$ untuk $\mathbf{y}_i = -1$

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix}$$

$$\mathbf{x}_i' = \begin{bmatrix} x_{i1} & x_{i2} & \dots & x_{ip} \end{bmatrix} \tag{2.7}$$

$$\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{bmatrix}$$

Berdasarkan pada Gambar 2.3 dapat diketahui bahwa **w** merupakan garis pemisah antar kedua klasifikasi, dimana panjang **w** adalah *norm* dari $\|\mathbf{w}\|$. Terdapat dua bidang pembatas, yaitu bidang pembatas pertama yang memebatasi kelas (+1) sedangkan bidang pembatas kedua yang membatasi kelas (-1). Bidang pembatas (+1) yaitu $\mathbf{x}_i'\mathbf{w} + b = 1$ dan bidang pembatas (-1) adalah $\mathbf{x}_i'\mathbf{w} + b = -1$. Bidang pembatas (+1) memiliki nilai bobot **w** dan jarak tegak lurus dari titik asal sebesar $\frac{|1-b|}{\|\mathbf{w}\|}$, sedangkan untuk bidang pembatas (-1) memiliki nilai bobot **w** dan tegak lurus dari

titik asal sebesar $\frac{|-1-b|}{\|\mathbf{w}\|}$. Sehingga nilai maksimum margin atau nilai jarak antar bidang pembatas adalah $\frac{1-b-(-1-b)}{\|\mathbf{w}\|}$ atau sama dengan $\frac{2}{\|\mathbf{w}\|}$. Secara matematis optimasi SVM untuk klasifikasi *Linear* dalam primal adalah $\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2$. Pada optimasi tersebut sama dengan memaksimalkan $\|\mathbf{w}\|$. Optimasi ini akan dilakukan dengan *Lagrange Multiplier*, dimana rumus *Lagrange Multiplier* adalah sebagai berikut.

$$L(\mathbf{w},b,\boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n} \alpha_i \left[y_i \left(\mathbf{x}_i'\mathbf{w} + b \right) - 1 \right] \quad (2.8)$$

dimana α_i merupakan *Lagrange Multiplier* yang bernilai nol atau positif. Nilai optimal dari persamaan 2.8 dapat dihitung dengan meminimalkan *L* terhadap **w** dan *b*, serta memaksimalkan *L* terhadap α_i sehingga diperoleh persamaan berikut.

$$\max_{\boldsymbol{\alpha}} L_d = \max_{\boldsymbol{\alpha}} \left(\min_{\mathbf{w},b} L(\mathbf{w},b,\boldsymbol{\alpha}) \right) \quad (2.9)$$

Meminimalkan *L* terhadap **w** dan *b* dapat dilakukan dengan cara berikut.

$$\frac{\partial L(\mathbf{w},b,\boldsymbol{\alpha})}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = 0$$

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i \quad (2.10)$$

$$\frac{\partial L(\mathbf{w},b,\boldsymbol{\alpha})}{\partial b} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = 0$$

Sehingga Persamaan (2.10) dapat disubstitusikan pada persamaan (2.9) dan menghasilkan hasil sebagaimana pada Persamaan (2.11) berikut.

$$\max_{\boldsymbol{\alpha}} L_d = \max \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i' \mathbf{x}_j \qquad (2.11)$$

sedangkan untuk memaksimalkan α_i pada Persamaan 2.11 adalah sebagai berikut.

$$\sum_{i=1}^{n} \alpha_i y_i = 0, \;\; \alpha_i \geq 0 \qquad (2.12)$$

nilai α_i dapat diperoleh dan nilai tersebut digunakan untuk mencari nilai **w.** Jika nilai $\alpha_i > 0$ atau sebuah titik data ke-i untuk setiap $y_i(\mathbf{x}'\mathbf{w} + b) = 1$. Selanjutnya optimasi dengan *Lagrange Multiplier* yang telah diselesaikan pada tahap sebelumnya dapat digunakan untuk melakukan klasifikasi dengan aturan klasifikasi sebagai berikut.

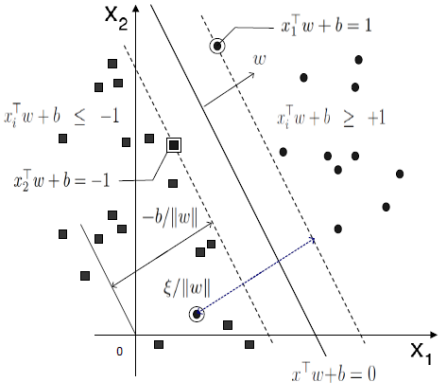
$$g(x) = sign(\mathbf{x}_i'\mathbf{w} + b) \qquad (2.13)$$

dimana $\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i x_i$ dan $b = -\frac{1}{2}(x_{+1} + x_{-1})w$ dengan x_{+1} dan x_{-1} adalah *support vector* yang mengikuti kelas berbeda untuk $y(\mathbf{x}'\mathbf{w} + b) = 1$. Nilai dari fungsi klasifikasi dapat dihitung sebagai berikut (Hardle, Prastyo, & Hafner, 2014).

$$f(x) = \mathbf{x}'\mathbf{w} + b \qquad (2.14)$$

2.5.2 *Support Vector Machine Linear Non-Separable*

Pada kasus *Linear* non-*separable* dapat diilustrasikan sebagaimana pada Gambar 2.3 berikut ini.



Gambar 2.3 Ilustrasi *Linear*y Non-*Separable* Klasifikasi SVM

(Sumber : Hardle, Prastyo, & Hafner, 2014)

Pada Metode SVM *Linear* non-*separable* terdapat variabel slack ξ_i *(soft margin*) yang menunjukan pelanggaran terhadap ketelitian pemisahan. Pelanggaran pemisahan ini memungkinkan terdapat suatu titik yang berada pada margin *error* atau yang biasa disebut dengan *miss classification*, sehingga $\mathbf{x}_i$ diklasifikasikan menjadi sebagai berikut.

$$\mathbf{x}_i'\mathbf{w} + b \ge 1 - \xi_i \text{ untuk } y_i = +1$$
$$\mathbf{x}_i'\mathbf{w} + b \le -(1 - \xi_i) \text{ untuk } y_i = -1$$
sehingga, (2.15)
$$y_i\left(\mathbf{x}_i'\mathbf{w} + b\right) \ge 1 - \xi_i$$
$$\xi_i \ge 0$$

Sehingga rumus untuk optimasi bidang pemisah menjadi sebagai berikut.

$$\min_{\mathbf{w},\xi} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \quad (2.16)$$

dimana nilai $\xi_i \geq 0$ dan parameter $C > 0$ dimana C (*cost*) adalah parameter yang menentukan besar biaya akibat *miss classification* dari data *training*. Apabila nilai C besar, maka margin akan menjadi lebih kecil, yang dapat mengindikasikan bahwa tingkat kesalahan akan menjadi lebih kecil. Sebaliknya, apabila nilai C kecil maka dapat diindikasikan bahwa tingkat kesalahan akan lebih besar. Fungsi *Lagrange Multiplier* pada kasus *Linear* non-*separable* adalah sebagai berikut.

$$L(\mathbf{w},b,\xi,\boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\alpha_i - \sum_{i=1}^{n}\alpha_i\left[y_i\left(\mathbf{x}_i'\mathbf{w}+b\right)-1+\xi_i\right] - \sum_{i=1}^{n}\mu_i\xi_i \quad (2.17)$$

dimana $\alpha_i \geq 0$ dan $\mu_i \geq 0$ adalah *Lagrange Multiplier*, $\alpha_i\left[y_i\left(\mathbf{x}_i'\mathbf{w}+b\right)-1+\xi_i\right]=0$ dan $\mu_i\xi_i = 0$. Sehingga nilai optimal dapat diperoleh dengan meminimalkan L terhadap **w**, b, dan ξ_i serta dengan memaksimumkan L terhadap α_i sehingga diperoleh persamaan sebagai berikut.

$$\max_{\boldsymbol{\alpha}} L_d = \max_{\boldsymbol{\alpha}}\left(\min_{\mathbf{w},b,\xi} L\left(\mathbf{w},b,\xi,\boldsymbol{\alpha}\right)\right) \quad (2.18)$$

Meminimalkan L terhadap **w**, b, dan ξ_i dapat dilakukan dengan cara berikut.

$$\begin{aligned}
\frac{\partial L\left(\mathbf{w},b,\xi,\boldsymbol{\alpha}\right)}{\partial \mathbf{w}} &= \mathbf{w} - \sum_{i=1}^{n}\alpha_i y_i \mathbf{x}_i = 0 \\
\mathbf{w} &= \sum_{i=1}^{n}\alpha_i y_i \mathbf{x}_i \\
\frac{\partial L\left(\mathbf{w},b,\xi,\boldsymbol{\alpha}\right)}{\partial b} &= \sum_{i=1}^{n}\alpha_i y_i \mathbf{x}_i = 0 \\
\frac{\partial L\left(\mathbf{w},b,\xi,\boldsymbol{\alpha}\right)}{\partial \xi_i} &= C - \alpha_i - \mu_i \\
\alpha_i &= C - \mu_i
\end{aligned} \quad (2.19)$$

Sehingga Persamaan (2.19) dapat disubstitusikan pada persamaan (2.18) dan menghasilkan fungsi sebagai berikut.

$$\max_{\alpha} L_d = \max_{\alpha} \left(\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \mathbf{x}_i' \mathbf{x}_j \right) \quad (2.20)$$

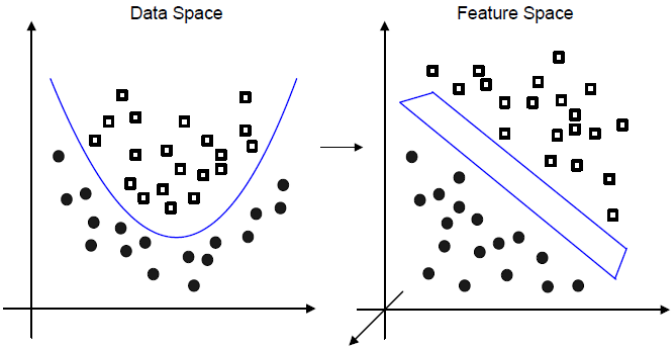
sedangkan untuk memaksimalkan α_i pada Persamaan 2.14 adalah sebagai berikut.

$$\sum_{i=1}^{n} \alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C \quad (2.21)$$

sampel $\mathbf{x}_i$ untuk $\alpha_i > 0$ (*support vector*) adalah titik yang berada di atas margin atau dalam margin ketika *soft* margin digunakan. *support vector* sering menyebar dan level penyebarannya berada pada batas atas (*upper bound*) untuk *miss classification rate*.

2.5.3 *Support Vector Machine Non-Linear Separable*

Pada umumnya kasus nyata sangat jarang sekali dijumpai data yang *Linear*, melainkan banyak data yang tidak *Linear*. Sehingga dikembangkan SVM untuk kasus non linier dengan memasukkan konsep *kernel* pada ruang fitur berdimendi yang lebih tinggi. SVM non *Linear separable* dapat diilustrasikan sebagai berikut.



Gambar 2.4 Ilustrasi Non *Linear Separable* Klasifikasi SVM

(Sumber : Hardle, Prastyo, & Hafner, 2014)

Pada Gambar 2.4 (kiri) menunjukan bahwa data pada kelas lingkaran dan kotak tidak dapat dipisahkan secara *Linear* jika menggunakan dimensi dua, sedangkan pada gambar sebelah kanan dapat dilihat bahwa ruang dengan dimensi tiga dapat memisahkan secara *Linear* oleh *hyperplane*. Untuk mengubah menjadi dimensi yang lebih tinggi, perlu dilakukan transformasi dengan menggunakan *kernel trick*. *Kernel trick* merupakan perhitungan *scalar product* melalui sebuah fungsi *kernel*. Sehingga fungsi *kernel* yang terbentuk adalah sebagai berikut.

$$K\left(\mathbf{x}_i,\mathbf{x}_j\right)=\psi\left(\mathbf{x}_i\right)'\psi\left(\mathbf{x}_j\right) \tag{2.22}$$

Sehingga persamaan optimasi pada Persamaan 2.20 berubah menjadi,

$$\max_{\boldsymbol{\alpha}} L_d=\max_{\alpha}\left(\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j K\left(\mathbf{x}_i,\mathbf{x}_j\right)\right) \tag{2.23}$$

sedangkan untuk memaksimalkan α_i pada Persamaan 2.23 adalah sebagai berikut.

$$\sum_{i=1}^{n}\alpha_i y_i=0,\;\; 0\le\alpha_i\le C \tag{2.24}$$

Nilai dari fungsi klasifikasi dapat dirumuskan sebagai berikut.

$$\begin{aligned} f\left(\mathbf{x}_i\right)&=\sum_{i=1}^{n}\alpha_i y_i\left(\psi\left(\mathbf{x}_i\right)'\psi\left(\mathbf{x}_j\right)\right)+b\\ &=\sum_{i=1}^{n}\alpha_i y_i K\left(\mathbf{x}_i,\mathbf{x}_j\right)+b \end{aligned} \tag{2.25}$$

Sedangkan fungsi *kernel* yang sering digunakan telah ditabelkan pada Tabel 2.1 sebagai berikut.

Tabel 2.1 Fungsi *Kernel* SVM

No	Nama *Kernel*	Fungsi *Kernel*
1.	*Linear*	$K\left(\mathbf{x}_i,\mathbf{x}_j\right)=\mathbf{x}_i'\mathbf{x}_j+C$
2.	*Polynomial Kernel*	$K\left(\mathbf{x}_i,\mathbf{x}_j\right)=\left(\delta\mathbf{x}_i'\mathbf{x}_j+r\right)^p,\gamma>0$
3.	*Radial Basis Function (RBF)*	$K\left(\mathbf{x}_i,\mathbf{x}_j\right)=\exp\left(-\gamma\left\|\mathbf{x}_i-\mathbf{x}_j\right\|^2\right),\gamma>0$
4.	*Sigmoid*	$K\left(\mathbf{x}_i,\mathbf{x}_j\right)=\tanh\left(\delta\mathbf{x}_i'\mathbf{x}_j+r\right)$

2.6 Regresi Logistik

Regresi logistik adalah suatu metode yang dapat digunakan untuk mencari hubungan antara variabel respon yang bersifat *dichotomous* (dua kategori) atau *polychotomous* (lebih dari dua kategori) dengan satu atau lebih variabel prediktor ber-skala kategori atau kontinu (Hosmer & Lemeshow, 2000). Model yang didapat dapat dijadikan model dalam mengklasifikasikan variabel prediktor ke dalam variabel respon yang berupa data kategorik. Anggap bahwa sekumpulan *p* variabel bebas ditunjukkan sebagai $\mathbf{x}=\left(x_1,x_2,\ldots,x_p\right)$. Bentuk logit dari regresi logistik multivariabel adalah sebagai berikut.

$$g(x)=\ln\left[\frac{\pi(x)}{1-\pi(x)}\right]=\beta_0+\beta_1x_1+\ldots+\beta_px_p \tag{2.26}$$

Selanjutnya untuk model regresi logistik dengan *p* adalah banyaknya variabel independen sebagaimana pada persamaan (2.13)

$$\pi(x)=\frac{e^{g(x)}}{1+e^{g(x)}}=\frac{\exp(\beta_0+\beta_1x_1+\ldots+\beta_px_p)}{1+\exp(\beta_0+\beta_1x_1+\ldots+\beta_px_p)} \tag{2.27}$$

2.6.1 Estimasi Parameter

Estimasi parameter dari model regresi logistik dapat dilakukan dengan menggunakan metode *Maximum Likelihood*

Estimation (MLE). Fungsi probabilitas distribusi *bernoulli* di setiap pengamatan (x_i, y_i) ditunjukkan pada persamaan berikut.

$$f(y_i) = \pi(x_i)^{y_i} \left[(1 - \pi(x_i))\right]^{1-y_i} \tag{2.28}$$

Apabila antar pengamatan diasumsikan independen, maka fungsi likelihood dari pengamatan yang independen adalah sebagai berikut.

$$l(\beta) = \prod_{i=1}^{n} f(y_i) = \prod_{i=1}^{n} \pi(x_i)^{y_i} (1 - \pi(x_i))^{1-y_i} \tag{2.29}$$

Fungsi *likelihoodl*(**β**)kemudian diubah ke persamaan ln.

$$L(\beta) = \ln l(\beta) = \sum_{j=0}^{p} \left[\sum_{i=1}^{n} y_i x_{ij} \right] \beta_j - \sum_{i=1}^{n} \ln \left[1 + \exp\left(\sum_{j=0}^{p} \beta_j x_{ij} \right) \right] \tag{2.30}$$

Selanjutnya *L*(**β**) dari persamaan 2.30 diturunkan terhadap β_j dan hasilnya sama dengan 0

$$\frac{\partial L(\beta)}{\partial \beta_j} = \sum_{i=1}^{n} y_i x_{ij} - \sum_{i=1}^{n} x_{ij} \left(\frac{\exp\left(\sum_{j=0}^{p} \beta_j x_j \right)}{1 + \exp\left(\sum_{j=0}^{p} \beta_j x_j \right)} \right) = 0 \tag{2.31}$$

Guna mengestimasi parameter β, digunakan smetode numerik, yaitu Metode iterasi *Newton Raphson*, sedangkan untuk estimasi varians dan kovarians, diperoleh dari turunan kedua fungsi ln *likelihood L*(**β**). Berdasarkan turunan kedua fungsi ln *likelihood*, dapat diperoleh matriks varians dan kovarians dari estimasi parameter melalui invers matriks (Agresti, 2007).

$$\text{cov}\left(\hat{\beta}\right) = \left\{ \text{X}' diag\left[\hat{\pi}_i \left(1 - \hat{\pi}_i\right) \right] \text{X} \right\}^{-1} \tag{2.32}$$

Dimana $diag\left[\hat{\pi}_i \left(1 - \hat{\pi}_i\right) \right]$ pada persamaan 2.32 merupakan $n \times n$ matriks diagonal dengan elemen diagonal utama

yaitu $\hat{\pi}(x_i)(1-\hat{\pi}(x_i))$ dimana akar kuadrat dari elemen-elemen diagonal utama adalah estimasi parameter model.

2.7 Performa Klasifikasi

Pengukuran performa dilakukan untuk melihat hasil yang didapatkan dari klasifikasi. Terdapat beberapa cara untuk mengukur performa, beberapa cara yang sering digunakan adalah dengan menghitung *accuracy*, *sensitivity*, dan *specificity* (Hotho, Nurnburger, & Paass, 2005) Akurasi merupakan persentase dari total dokumen yang teridentifikasi secara tepat dalam proses klasifikasi. Ketiga pengukuran performa tersebut dapat diperoleh dari perumusan 2.19 dengan *confusion matrix* sebagaimana pada Tabel 2.2 berikut

Tabel 2.2 *Confusion Matrix*

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

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$
$$precision = \frac{TP}{TP+FP} \quad (2.33)$$
$$recall = \frac{TP}{TP+FN}$$

Dimana TP adalah *True Positive*, FP adalah *False Positive*, TN adalah *True Negatif*, dan FN adalah *False Negatif*. Nilai *accuracy*, sensitivity, dan specificity yang telah diperoleh maka dapat dicari pengukuran ketepatan klasifikasi lebih lanjut menggunakan perhitungan *Area Under Curve* (AUC) (Chawla, 2005). Perhitungan *Area Under Curve* (AUC) dapat dilakukan sebagaimana persamaan 2.34 berikut.

$$AUC = \frac{1}{2}\left(\frac{TP}{TP+FN}+\frac{TN}{TN+FP}\right) \quad (2.34)$$

Adapun selang ketepatan klasifikasi yang baik dengan perhitungan AUC dapat dilihat pada Tabel 2.3 berikut ini (Bekkar, Djemaa, & Alitouche, 2013).

Tabel 2.3 Interpretasi Nilai AUC

Nilai AUC	Keterangan
0.5-0.6	Kurang
0.6-0.7	Cukup
0.7-0.8	Baik
0.8-0.9	Sangat Baik
0.9-1.0	Sempurna

2.8 *Synthetic Minority Oversampling TEchnique* (SMOTE)

Data imbalance merupakan data yang tidak diinginkan oleh suatu penelitian, oleh sebab itu diperlukan metode untuk menanganinya. Dalam data imbalance terdapat perbedaan jumlah klass yang signifikan, dimana kelas yang dominan disebut dengan mayoritas dan kelas yng lebih sedikit disebut dengan minoritas. Synthetic Minority *Oversampling* TEchnique (SMOTE) merupakan algoritma yang diperkenalkan pertama kali oleh Nithes V Chawla pada tahun 2002. Metode ini merupakan salah satu metode *oversampling* yang mana dilakukan dengan penerapan metode sampling. Metode sampling disini dilakukan untuk meningkatkan jumlah kelas minoritas melalui replikasi data secara acak. Replikasi ini dilakukan dengan pemilihan k nearest neighbor sebagai penentuannya, sehingga jumlah data yang minoritas seimbang dengan data mayoritas. Secara umum dapat dirumuskan sebagai berikut (Sain H. & Purnami Santi Wulan, 2015).

$$x_{syn} = x_i + \left(x_{knn} - x_i\right)\delta \tag{2.35}$$

Dimana i = 1, 2, 3, …, jumlah kata kunci dan δ adalah angka acaka antara 0 dan 1. Algoritma untuk melakukan metode SMOTE dapat dijelaskan berdasarkan pada Gambar 2.5 berikut ini.

Algorithm *SMOTE*(T, N, k)
Input: Number of minority class samples T; Amount of SMOTE N%; Number of nearest neighbors k
Output: $(N/100) * T$ synthetic minority class samples
1. (∗ *If N is less than 100%, randomize the minority class samples as only a random percent of them will be SMOTEd.* ∗)
2. **if** $N < 100$
3. **then** Randomize the T minority class samples
4. $T = (N/100) * T$
5. $N = 100$
6. **endif**
7. $N = (int)(N/100)$ (∗ *The amount of SMOTE is assumed to be in integral multiples of 100.* ∗)
8. k = Number of nearest neighbors
9. $numattrs$ = Number of attributes
10. $Sample[\,][\,]$: array for original minority class samples
11. $newindex$: keeps a count of number of synthetic samples generated, initialized to 0
12. $Synthetic[\,][\,]$: array for synthetic samples
(∗ *Compute k nearest neighbors for each minority class sample only.* ∗)
13. **for** $i \leftarrow 1$ **to** T
14. Compute k nearest neighbors for i, and save the indices in the $nnarray$
15. Populate(N, i, $nnarray$)
16. **endfor**

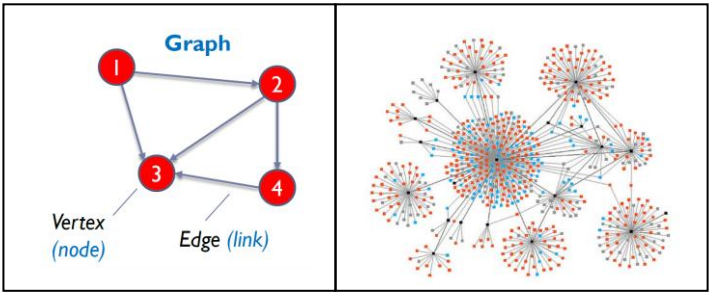
Populate(N, i, $nnarray$) (∗ *Function to generate the synthetic samples.* ∗)
17. **while** $N \neq 0$
18. Choose a random number between 1 and k, call it nn. This step chooses one of the k nearest neighbors of i.
19. **for** $attr \leftarrow 1$ **to** $numattrs$
20. Compute: $dif = Sample[nnarray[nn]][attr] - Sample[i][attr]$
21. Compute: gap = random number between 0 and 1
22. $Synthetic[newindex][attr] = Sample[i][attr] + gap * dif$
23. **endfor**
24. $newindex{+}{+}$
25. $N = N - 1$
26. **endwhile**
27. **return** (∗ *End of Populate.* ∗)
End of Pseudo-Code.

Gambar 2.5 Algoritma SMOTE
(Chawla, 2005)

2.9 *Social Network Analysis* (SNA)

Social Network Analysis (SNA) merupakan studi yang mempelajari tentang hubungan dengan memanfaatkan teori *graph*. *Social Network Analysis* berfokus pada pola hubungan antar aktor serta menggambarkan hubungan jaringan tersebut. *Social*

Network Analysis dapat digunakan untuk mempelajari pola jaringan organisasi, ide-ide, dan orang-orang yang terhubung melalui berbagai cara dalam sebuah lingkaran. Berikut visualisasi dari *graph Social Network Analysis*.



Gambar 2.6 Graph *Social Network Analysis*

(Sumber: Cheliotis, 2010)

Ukuran (*metric*) yang digunakan dalam penentuan aktor dalam penelitian ini adalah *degree centrality*, *betweenness centrality*, dan *closeness centrality*.

1. Degree Centrality

Degree centrality merupakan jumlah *links* pada *node*, *degree centrality* juga berguna dalam menilai *node* yang *central* untuk menyebarkan informasi dan mempengaruhi orang lain dalam lingkungan mereka. Dalam sebuah jaringan dengan *node*, *degree centrality* dari *node* ini adalah sebagai berikut.

$$C_D(n_i) = d(n_i) \tag{2.36}$$

Dimana, $d(n_i)$ adalah banyaknya *link* atau garis yang terkait pada jaringan. Suatu *node* mempunyai nilai *degree centrality* antara 0 hingga *g-1*. Untuk membedakan *centrality* dalam *node* dari jaringan dengan skala yang berbeda, normalisasi *degree centrality* diperlukan sehingga dapat dirumuskan sebagai berikut.

$$C_D^{'}(n_i) = \frac{d(n_i)}{g-1} \tag{2.37}$$

2. Closeness Centrality

Closeness adalah rata-rata semua jalur terpendek dari satu *node* untuk semua *node* yang lainnya dalam jaringan. Ukuran dari jangkauan, yaitu berapa lama akan mencapai *nodes* lain dari sebuah *node* awal. *Closeness centrality* berguna dalam kasus dimana kecepatan penyebaran informasi adalah perhatian utama. Dalam sebuah jaringan dengan *g nodes*, *closeness centrality* dari *node* n_i diperoleh persamaan berikut.

$$C_c(n_i) = \left[\sum_{j=1}^{g} d(n_i, n_j)\right]^{-1} \tag{2.38}$$

Dimana, $d(n_i, n_j)$ adalah jumlah *edge* yang terhubung dari n_i dan n_j. Sedangkan untuk membandingkan di jaringan dengan skala yang berbeda, *closeness centrality* di normalisasikan dengan mengalikan *g-1*, sehingga diperoleh persamaan sebagai beikut.

$$\begin{aligned} C_c^{'}(n_i) &= \frac{g-1}{\left[\sum_{j=1}^{g} d(n_i, n_j)\right]} \\ &= (g-1)C_c(n_i) \end{aligned} \tag{2.39}$$

3. Betweenness Centrality

Betweenness centrality adalah jumlah jalur terpendek yang melewati sebuah *node* dibagi dengan semua jalur terpendek dalam jaringan. *Betweenness centrality* menunjukkan *node* yang lebih cenderung berada dalam jalur komunikasi antara *node* lain. Dalam satu jaringan dengan *g nodes*, *betweenness centrality*

untuk *node* didefinisikan sebagai jumlah dari jalur terpendek yang melalui *node* pada persamaan berikut.

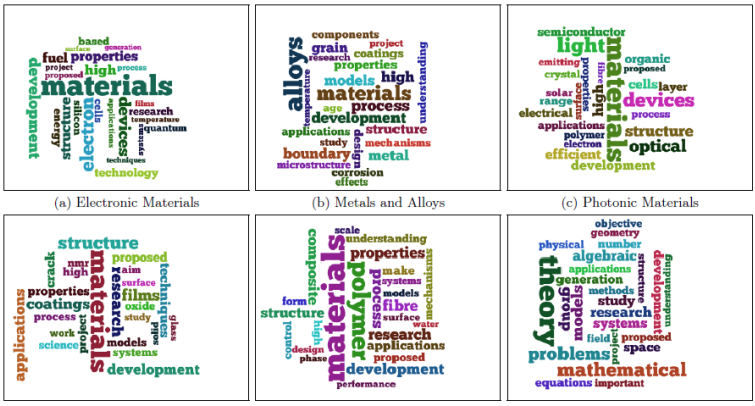
$$C_b(n_i) = \frac{\sum_{j<k} g_{jk}(n_i)}{g_{jk}} \tag{2.40}$$

Dimana, g_{jk} adalah jumlah jalur terpendek antara 2 *node* dalam jaringan g_{jk}, n_i adalah jumlah jalur terpendek dari *node j* ke *node k* melewati *node i*, dengan normalisasi dapat dirumuskan sebagaimana pada persamaan berikut.

$$C_b^{'}(n_i) = \frac{C_b^{'}(n_i)}{[(g-1)(g-2)/2]} \tag{2.41}$$

2.10 *Word Cloud*

Word Cloud merupakan salah satu metode yang sering digunakan untuk penggambaran data teks. *Word Cloud* dapat melakukan representasi sebuah data teks dengan cara membuat *plot* kata-kata yang sering muncul. Semakin sering kata itu muncul maka huruf kata tersebut semakin besar, begitu juga apabila suatu kata jarang muncul maka ukuran kata itu akan lebih kecil dari yang lainnya. Berikut ini contoh dari penggambaran data teks dengan *Word Cloud* ditunjukkan pada Gambar 2.7 (Castella & Sutton, 2014).



Gambar 2.7 *Word Cloud*
(Sumber: Castella & Sutton, 2014)

2.11 Twitter

Media sosial merupakan perkembangan teknologi *web* yang berbasis pada internet, yang memudahkan semua orang untuk berkomunikasi, saling berbagi, dan membentuk sebuah jaringan secara *online*. Terdapat berbagai jenis media sosial yang telah terkenal salah satunya adalah *twitter*. *Twitter* adalah suatu situs *web* yang merupakan layanan dari *microblog*, yaitu *blog* yang memberikan fasilitas bagi pengguna untuk dapat menuliskan pesan. Akan tetapi pesan yang hendak di *posting* memiliki batasan jumlah karakter yaitu 140 karakter. *Twitter* pertama kali ada pada tanggal 21 Januari 2000 di San Fransisco, California. *Twitter* merupakan jejaring sosial besar yang fokus pada kecepatan komunikasi. Kecepatan dan kemudahan dalam hal publikasi pesan membuat *twitter* menjadi media komunikasi yang digemari. Pengguna dapat terhubung dengan pengguna lain melalui fitur '*follow*', sehingga pengguna dapat mengikuti *tweet* terbaru dari pengguna yang di *follow*. Hal-hal lain yang dapat dilakukan dengan jejaring sosial twitter adalah RT atau *retweet* yang merupakan sarana membalas *tweet* dengan menyertakan isi *tweet* sumber, sehingga pengguna yang menerima *retweet* bisa

memahami konteks pesan yang diterima. Selain itu ada pula *hastag* atau dengan simbol '#' yang diikuti sebuah kata untuk menandai konteks dari sebuah pesan *twitter*, namun *hastag* bukanlah syarat publikasi *tweet*. Adapun beberapa alasan *tweet* digunakan sebagai sumber penelitian, diantaranya adalah frekuensi *posting* pesan yang sangat tinggi, pesan *twitter* tidak terlalu panjang sehingga lebih deskriptif dan mudah dimengerti dan dapat cepat diklasifikasikan (Setyani, 2013).

2.12 Pemerintah Kota Surabaya

Surabaya adalah kota terbesar dan tertua di Indonesia, dengan total luas 330,45 km^2 dan jumlah penduduk lebih dari 3 juta orang di malam hari dan lebih dari 5 juta orang di jam kerja. Surabaya terletak di timur laut Pulau Jawa. Surabaya juga dikenal sebagai kota pahlawan, gelar itu diberikan terkait dengan semangat heroik dan memperingati pertempuran surabaya pada tanggal 10 November 1945. Menurut web resmi Surabaya di https:// sparkling.surabaya.go.id/ about-surabaya/the-history-of-surabaya/ Surabaya merupakan kota yang memiliki amanah untuk menuju makmur, yang artinya Surabaya harus bisa memberikan kemakmuran, kesehatan, keselamatan dan kedamaian untuk warga atau masyarakat untuk berpartisipasi aktif dalam pembangunan. Surabaya juga merupakan kota jasa dan perdagangan yang berarti kota yang mendasarkan kegiatannya pada pembangunan ekonomi dengan berfokus pada karakteristik orang-orang kota. Hal ini juga termasuk pelayanan yang menjadi tulang punggung pembangunan dalam rangka mewujudkan kesejahteraan masyarakat dengan memperhatikan potensi lokal. Surabaya juga menjadi pusat layanan internasional dan perdagangan yang didukung dengan akses cukup ke sumber daya produktif, tata pemerintahan yang baik, infrastruktur kota terpadu dan efisien dan fasilitas, serta mampu meningkatkan ekonomi lokal, produk dan inovasi layanan, dan pengembangan layanan dan industri kreatif global yang kompetitif.

Surabaya mampu mempertahankan kemampuan untuk mengintegrasikan proses perkembangan pesat dengan tetap memperhatikan kelestarian lingkungan akun dan kapasitasnya membawa melalui perbaikan fasilitas umum dan infrastruktur kota yang ramah lingkungan ramah. Kalimas merupakan sungai terbesar membelah kota Surabaya. Di masa lalu, itu adalah pusat kegiatan ekonomi. Sebagai kota berkembang, aktivitas ekonomi tidak lagi terpusat di Kalimas. Dengan demikian, daerah ini diabaikan dan menjadi daerah kumuh. Upaya revitalisasi besar dilakukan mengetahui pentingnya sungai dalam pembangunan kota. Poin revitalisasi potensi dikembangkan termasuk pembangunan daerah monumen kapal selam. Dalam area ini, ikon dari Surabaya yang disebut "Suro dan Boyo" patung di tinggi dibangun 15 meter. Sepanjang tepi sungai, ada BMX dan Skate Park sebagai titik pertemuan utama skaters Surabaya, Jayengrono Park, Prestasi Park, Expression Park, dan Food Court Ketabang Kali. Kalimas sebelumnya diabaikan dan kini daerah kumuh yang sekarang berubah menjadi tempat yang menarik untuk dikunjungi dan memperkenalkan konsep waterfront city yang menambah pesona Surabaya. Untuk mengatasi kekurangan sumber daya manusia dan memberikan layanan berkualitas tinggi untuk warga, Kota memanfaatkan penggunaan teknologi dalam memberikan layanan publik. Denga tidak terbatasnya anggaran pada perencanaan dan pengawasan, pengadaan, pendidikan, pelayanan kesehatan, mengeluarkan izin, pajak, pengawasan keamanan, dll. Selain itu, Surabaya juga menyediakan tempat publik utama dengan akses jaringan internet untuk memudahkan warga dalam mengakses dan memanfaatkan teknologi. Untuk lebih memperluas kemudahan penggunaan dan portabilitas, Surabaya juga mengoptimalkan penggunaan aplikasi mobile dalam memberikan pelayanan publik. Di bidang pendidikan, Hukum Pendidikan Nasional nomor 20 tahun 2003 pasal 5 ayat (1) menyebutkan bahwa setiap warga negara memiliki hak yang sama terhadap pendidikan berkualitas. Dengan demikian, Pemerintah Kota Surabaya berkomitmen untuk memberikan pendidikan

berkualitas untuk semua orang, tidak hanya di sekolah negeri tapi juga swasta. Para siswa miskin berprestasi diberikan beasiswa untuk melanjutkan studi ke tingkat pendidikan tinggi. Dalam rangka mendukung peningkatan kapasitas dan peningkatan kualitas guru, beasiswa diberikan untuk mengejar gelar sarjana dan pascasarjana. Pada tahun 2013, pemerintah kota mengalokasikan anggaran untuk gelar kesetaraan sarjana dan pascasarjana tingkat persamaan. Surabaya telah diakui secara internasional dalam banyak aspek karena beberapa prestasi dan penghargaan seperti Asian Townscape Award oleh PBB; ASEAN Environment Sustainable City Award; Asian Cities of the Future.

2.13 Radio Suara Surabaya

Suara Surabaya FM 100,55 Mhz, mengudara bersamaan dengan momentum gerhana matahari total pada tanggal 11 Juni 1983 dari lokasi di kawasan berbukit jalan wonokitri besar 40. Suara Surabaya merupakan radio pertama di indonesia yang sejak awal kelahirannya secara sadar menerapkan format ‘radio news atau informasi’ dan bermotto fm news & musik hit. Menyadari konsekuensi dari penerapan format ‘radio news atau informasi’ dari awal kelahirannya, yaitu dengan membentuk ‘tim redaksi’ tahun 1983, kemudian mengembangkan 'tim reporter' pada tahun 1987, dan ketika tahun 1995 dikembangkan konsep interaktif maka muncul 'tim redaksi interaktif atau gate keeper' dengan motto news, interaktif, dan solutif.

Tahun 2000 suara surabaya mengembangkan diri memasuki era *cyberspace* yaitu suarasurabaya.net. Akhir 2002, meluncurkan majalah bulanan mossaik, tentang informasi dan gaya hidup. Pada pertengahan 2003 Giga FM 99.85 Mhz menjadi bagian dari Suara Surabaya FM. Tepat pada tanggal 3 Mei 2004 pukul 00.00 WIB, radio Suara Surabaya FM 100.55 Mhz berpindah kanal frekuensi menjadi radio Suara Surabaya FM 100 mhz. Giga FM 99.85 Mhz juga turut berpindah kanal frekuensi baru yaitu Giga FM 99.6 Mhz.

Radio Suara Surabaya memiliki visi menjadi sumber pemberdayaan dan kegiatan demokratisasi masyarakat, melalui usaha kegiatan media massa yang mengikuti perkembangan teknologi komunikasi dan telekomunikasi. Adapun misi dari radio suarabaya yang pertama adalah suara surabaya, perusahaan media massa yang dituntut berkembang dengan mengandalkan kemajuan teknologi komunikasi dan telekomunikasi. Yang kedua adalah menjadi sentra informasi tentang Surabaya dan jawa timur. Misi selanjutnya adalah suara Surabaya menyelenggarakan berbagai kegiatan pemberdayaan proses demokratisasi di masyarakat, dan misi terakhir adalah menjadi sumber kehidupan dan kesejahteraan seluruh unsur karyawan yang bekerja untuk kemajuan bersama (Media, 2018).

(Halaman ini sengaja dikosongkan)

BAB III
METODOLOGI PENELITIAN

3.1 Sumber Data

Data yang digunakan dalam penelitian ini adalah sekumpulan *tweet* yang diperoleh dari pengguna twitter di Indonesia yang diambil pada tanggal 28 September 2017 sampai dengan 7 Mei 2018. Data diperoleh dengan menggunakan Twitter API (*Application Programming Interface*) pada akun resmi Pemerintah Kota Surabaya (@SapawargaSby) dan Radio Suara Surabaya (@e100ss).

3.2 Struktur Data dan Variabel Penelitian

Bedasarkan sumber data yang diperoleh berupa data *tweet*, maka data tersebut harus dilakukan *filter* dengan hanya mengambil data yang mengandung sentimen. Berikut variabel penelitian yang digunakan.

Tabel 3.1 Variabel Penelitian

Variabel	Keterangan	Skala Data
Y	Sentiment (Positif/Negatif) 0 = Sentiment Negatif 1 = Sentimen Positif	Nominal
X	Frekuensi kata ke-i yang muncul pada objek (twitter)	Rasio

Struktur data yang digunakan dalam penelitian ini setelah dilakukan praproses data teks *tweet* terdiri dari variabel respon (y) dan variabel prediktor (x). Dalam analisis sentiment ini, bentuk variabel prediktor yang digunakan adalah nilai dari hasil pembobotan sehingga dinotasikan dengan *w*. Berikut merupakan struktur data penelitian sebelum dan setelah dilakukan praproses data *tweet*.

Tabel 3.2 Struktur Data Penelitian Sebelum Praproses Data

No.	Nama *Account*	*Tweet* (*y*)	Klasifikasi Sentimen
1	Pemkot Surabaya @SapawargaSby	Ini dong juga di bersihkan @SapawargaSby	Negatif
.		.	.
.		.	.
.		.	.
n		@SapawargaSby Kok seperti yang di jakarta itu ya, jualan di tengah jalan	Negatif
1	Radio Suara Surabaya @e100ss	@e100ss tolong ditertibkan,saling srobot..macet dah 2 Jam https://t.co/npOizaFyFF	Negatif
.		.	.
.		.	.
.		.	.
m		"@e100ss lihat strobo polisi. Jalan macet tidak ada yang urgent main nyalain dan minta jalan. Tolong di infokan ke kom… https://t.co/CoVRO7HS0q	Negatif

Tabel 3.3 Struktur Data Penelitian Setelah Praproses Data

No.	Nama *Account*	Klasifikasi Sentimen (y)	Kata Kunci (w_1)	...	Kata Kunci (w_i)
1	Pemkot Surabaya @SapawargaSby	Negatif	$w_{1,1}$	...	$w_{1,I}$
2		Negatif	$w_{2,1}$	...	$w_{2,I}$
.		.	.	...	.
.	Radio Suara Surabaya @e100ss	.	.	...	.
.		.	.	...	.
$n+m$		Negatif	$w_{n+m,1}$	...	$w_{n+m,I}$

Berbeda halnya dengan struktur data dan variabel yang digunakan pada analisis *Social Network Analysis* (SNA). Pada analisis ini variabel yang dibutuhkan adalah *ID*, *Lebel*, *Source*, dan *Target. ID* merupakan Identitas dari setiap akun yang ada pada data, sedangkan *Label* adalah nama akun tersebut. Selanjutnya *Source* adalah *ID* dari akun yang melakukan *tweet* dan *Target* adalah ID dari akun tujuan *tweet* tersebut. Struktur data dari SNA dapat digamabrkan sebagaimana pada Tabel 3.4 berikut ini.

Tabel 3.4 Struktur Data *Social Network Analysis*

Node		***Edge***	
ID	***Label* (Nama Akun)**	***Source***	***Target***
1	112	25014	7980
2	80310	13440	7980
⋮	⋮	⋮	⋮
13487	Jusi_noviawati	10370	7980
⋮	⋮	⋮	⋮
26973	ZZehat	12813	25422
26974	Zzz_hanif	22047	4125

3.3 Langkah Analisis

Langkah analisis yang digunakan pada penelitian adalah sebagai berikut.

1. Mengambil data *tweet* dengan menggunakan Twitter API.
   a. Memasukkan *keyword* yang berhubungan dengan SapawargaSby dan e100ss.
   b. Menyimpan hasil *crawling* data dari kedua *keyword.*
2. Menyiapkan kata dasar yang diambil dari Kamus Besar Bahasa Indonesia untuk tahap *stemming* dan data daftar *stopwords* didapatkan dari tesis F. Z. Tala yang berjudul "*A Study of Stemming effect on Information Retrieval in Bahasa Indonesia".*
3. Melakukan klasifikasi sentiment secara manual untuk mengelompokan data *tweet* menjadi kategori positif atau negatif.

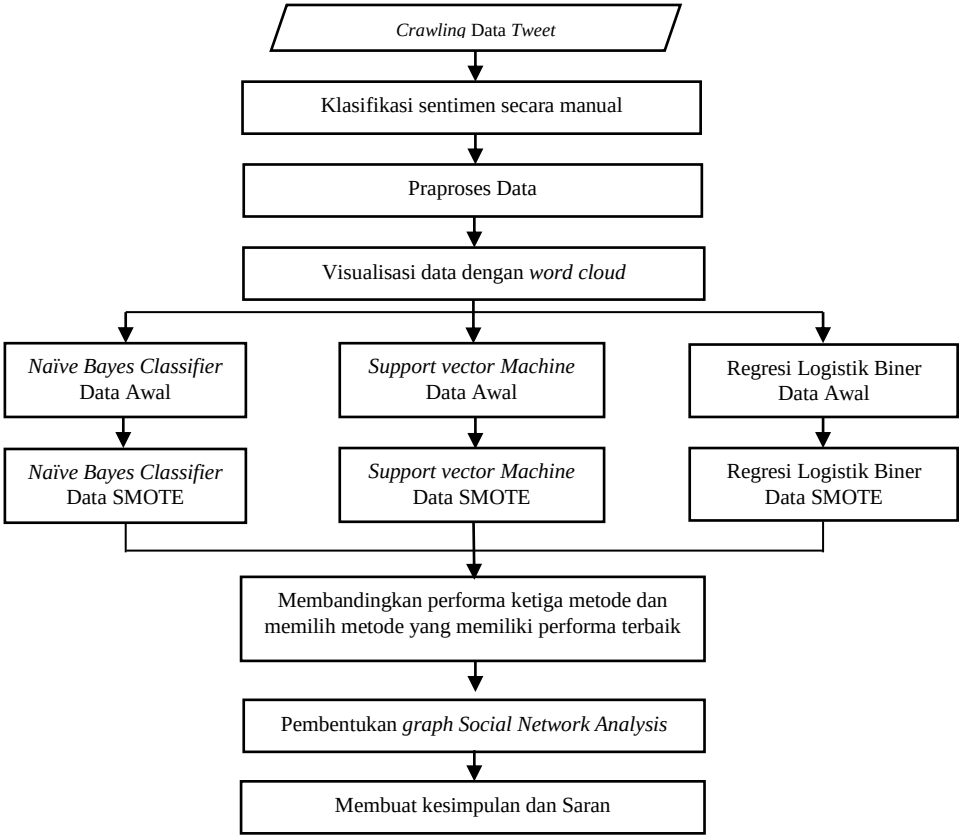
4. Melakukan praproses data.
   a. Menghapus *tweet* yang tidak mengandung sentimen (positif atau negatif).
   b. Melakukan *case folding*, yaitu mengubah semua teks dengan huruf kecil (non kapital) serta menghilangkan tanda baca.
   c. Melakukan *cleansing*, yaitu proses pembersihan *tweet* dari *noise*. Kata yang dihilangkan dalam *twitter* adalah karakter HTML, *emoticons*, *hastag (#)*, *username (@username)*, simbol *retweet* (*response tweet*) "RT", dan *link* URL.
   d. Melakukan *tokenizing* untuk memecah *tweet* menjadi kata per kata.
   e. Melakukan *stemming* menggunakan algoritma *confix-stripping stemmer* untuk mendapatkan kata dasar. Penentuna kata dasarnya berdasarkan daftar yang telah disiapkan.
   f. Melakukan proses *stopping* berdasarkan *stoplist* yang berisi *stopwords* yang telah ditentukan sebelumnya. Kata-kata yang terdapat pada *tweet* akan dibandingkan dengan daftar *stopwords*, jika terdapat kata-kata yang terdapat pada *stopwords* maka kata tersebut akan dihapus dari *tweet* sehingga ditemukan kata kunci yang identik.
   g. Mengubah data *tweet* kedalam bentuk frekuensi kemunculan kata mengguankan TF-IDF
5. Melakukan visualisasi *tweet* dengan *Word Cloud*
6. Klasifikasi data menggunakan *Naïve Bayes Classifier*
   a. Menghitung probabilitas tertinggi dari kategori sentimen yang diujikan (V_{MAP}).
   b. Mencari nilai V_{MAP} paling maksimum dan memasukkan *tweet* tersebut pada kategori dengan V_{MAP} maksimum.
   c. Menghitung performa klasifikasi dari model yang terbentuk.
   d. Melakukan langkah pada *point* a sampai c pada data SMOTE
7. Klasifikasi data menggunakan *Support Vector Machine*
   a. Menentukan parameter optimum pada SVM tiap jenis *kernel*

b. Memilih jenis *kernel* optimum berdasarkan performa klasifikasi berdasarkan parameter yang telah diperoleh
c. Melakukan langkah pada *point* a dan b pada data SMOTE

8. Klasifikasi data menggunakan Regresi Logistik
   a. Menentukan parameter optimum
   b. Menghitung performa klasifikasi dari model yang terbentuk berdasarkan parameter yang telah diperoleh
   c. Melakukan langkah pada *point* a dan b pada data SMOTE
9. Menentukan metode terbaik
   Membandingkan performansi metode NBC, SVM, dan Regresi logistik berdasarkan nilau *Accuracy Under Curve* (AUC) pada data awal dan data SMOTE
10. Melakukan *Social Network Analysis* untuk menentukan pola jaringan-jaringan antar pemilik akun yang berpendapat mengenai performa pemerintah kota Surabaya menggunakan *soft ware* Gephi
11. Membuat kesimpulan dan memberikan informasi tambahan untuk pemerintah kota Surabaya.

3.4 Diagram Alir

Langkah analisis sebagaimana telah dijelaskan pada sub bab sebelumnya dapat digambarkan dengan diagram alir sebagaimana pada Gambar 3.1 berikut.



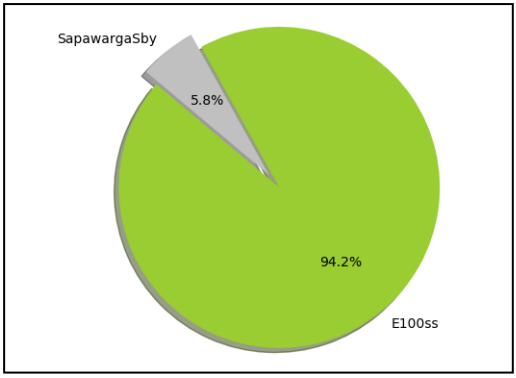
Gambar 3.1 Diagram Alir Penelitian

BAB IV
HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas mengenai hasil analisis data *tweet* masyarakat terhadap Pemerintah kota Surabaya (Pemkot Surabaya). Data yang digunakan merupakan data gabungan dari akun resmi Pemkot Surabaya (@SapawargaSby) dan akun resmi Radio Suara Surabaya (@e100ss) dengan jumlah data 1687 *tweet*. Akan dicari metode klasifikasi terbaik yang kemudian digunakan dalam analisis-analisis selanjutnya. Sebelum menganalisis, dilakukan praprosesing data terlebih dahulu.

4.1 Karakteristik Data *Tweet* Masyarakat terhadap Pemerintah Kota Surabaya

Sebelum melakukan praprosesing data, dilakukan analisis karakteristik data terlebih dahulu, dimana akan diketahui karakteristik data dari data *tweet* masyarakat terhadap Pemkot Surabaya. Data *tweet* masyarakan terhadap Pemkot Surabaya diperoleh dari dua akun resmi yaitu akun resmi Pemkot Surabaya (@SapawargaSby) dan akun resmi dari Radio Suara Surabaya. Berikut perbandingan sumber data yang digunakan.

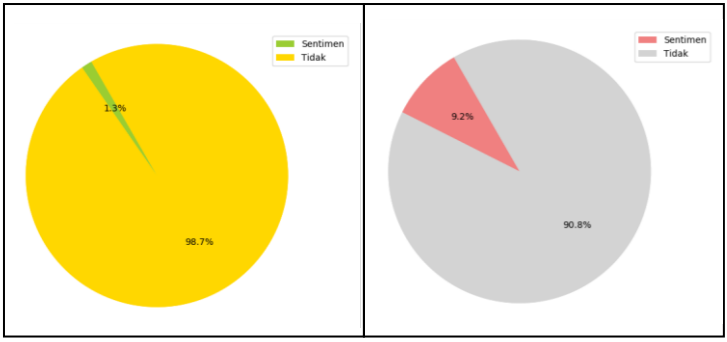


Gambar 4.1 Perbandingan Sumber Data

Berdasarkan Gambar 4.1 dapat diketahui bahwa 94,2% data diperoleh dari akun resmi Radio Suara Surabaya sedangkan 5,8%

sisanya dari akun resmi Pemkot Surabaya. Hal ini menunjukan bahwa masyarakat lebih aktif mengekspresikan keluh-kesahnya pada akun Radio Suara Surabaya. Sedangkan masyarakat yang melakukan *tweet* kepada akun resmi Pemkot Surabaya jauh lebih sedikit.

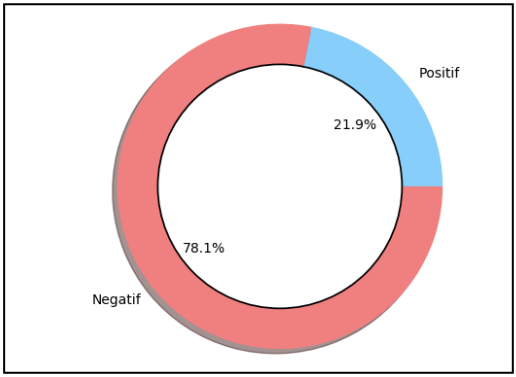
Dari data yang telah terkumpul tersebut kemudian diseleksi, hanya *tweet* yang mengandung sentimenlah yang akan dilakukan analisis lebih lanjut. Berikut perbandingan jumlah keseluruhan *tweet* dengan jumlah *tweet* yang mengandung sentimen.



Gambar 4.2 Jumlah Data Mengandung Sentimen @e100ss (kiri) @Sapawargasby (kanan)

Berdasarkan pada Gambar 4.2 dapat diketahui bahwa hanya 1,3% *tweet* dari akun @e100ss yang mengandung sentimen. Serta hanya 9,2% *tweet* dari akun @SapawargaSby yang mengandung sentimen. Hal ini diakrenakan mayoritas *tweet* yang ditujukan kepada akun akun terpilih ini berupa informasi, dimana informasi tidak masuk dalam sentimen.

Data yang mengandung sentimen kemudian dibagi menjadi dua klasifikasi yaitu positif dan negatif. Dari hasil klasifikasi positif dan negatif tersebut dapat diketahui perbandingan frekuensinya sebagai berikut.



Gambar 4.3 Karakteristik Data *Tweet*

Gambar 4.3 menunjukan bahwa dari total data 1687 *tweet* yang mengandung sentimen, sebesar 78,1% *tweet* masyarakat terhadap Pemkot Surabaya mengandung sentimen negatif dan sisanya sebesar 21,9% mengandung sentimen positif. Sehingga dapat disimpulkan bahwa mayoritas masyarakat memberikan keluhan kepada Pemkot Surabaya. Dari keluhan-keluhan tersebut akan diketahui konten yang sering muncul akan dijelaskan pada sub bab selanjutnya.

4.2 Praproses Data *Tweet* Masyarakat terhadap Pemerintah Kota Surabaya

Data *tweet* masyarakat terhadap Pemkot Surabaya yang telah dikumpulkan kemudian dilakukan filter dengan mengambil data-data *tweet* yang mengandung sentimen. Data yang mengandung sentimen kemudian dilakukan praroses data dengan tahap *cleansing*, *case folding*, *stemming*, *stopwords*, dan *tokenizing*. Berikut dilakukan praproses data pada data *tweet* dari salah satu akun yang ditujukan kepada akun resmi Suara Surabaya yaitu @e100ss. Data hasil simulasi praproses ditunjukan sebagaimana pada Tabel 4.1 berikut.

Tabel 4.1 Praproses Data

Data *tweet*	@e100ss tolong ditertibkan,saling srobot..macet dah 2 Jam https://t.co/npOizaFyFF
Menghapus *username* dan *link URL*	tolong ditertibkan,saling srobot..macet dah 2 Jam
Melakukan *Case folding*	tolong ditertibkan,saling srobot..macet dah 2 jam
Melakukan s*temming* dan menghilangkan *punctuation*	tolong tertib saling srobot macet dah 2 jam
Menghapus Stopwords	tolong tertib srobot macet jam

Dilakukan juga praproses data pada *tweet* "@e100ss lihat strobo polisi. Jalan macet tidak ada yang urgent main nyalain dan minta jalan. Tolong di infokan ke kom… https://t.co/CoVRO7HS0q". dengan langkah praproses yang sama dengan *tweet* sebelumnya, maka diperoleh hasil sebagaimana pada Tabel 4.2 berikut.

Tabel 4.2 Praproses Data Ke-2 Simulasi

Hasil Praproses Data	lihat strobo polisi jalan macet tidak urgent main nyalain minta jalan tolong kom

Langkah selanjutnya adalah perhitungan frekuensi masing-masing kata disetiap *tweet*. Dimana masing-masing kata merupakan variabel prediktor. Hasil dari perhitungan frekuensi kata adalah sebagai berikut.

Tabel 4.3 Hasil Perhitungan Frekuensi Kata Data Simulasi

Tweet	Variabel Prediktor					
	jalan	jam	…	tidak	tolong	urgent
1	0	1	…	0	1	0
2	2	0	…	1	1	1

Berdasarkan Tabel 4.3 dapat diketahui frekuensi kemunculan suatu kata pada sebuah *tweet*. Dapat diketahui bahwa kata "jalan" tidak muncul pada data *tweet* pertama akan tetapi muncul dua kali

pada data *tweet* kedua. Sedangkan untuk kata "Jam" muncul satu kali pada data *tweet* pertama dan tidak muncul pada data *tweet* kedua, begitu seterusnya.

Telah dilakukan praproses data pada seluruh data *tweet* masyarakat terhadap Pemkot Surabaya yang memberikan hasil sebagai mana pada Tabel 4.4 berikut ini.

Tabel 4.4 Hasil Perhitungan Frekuensi Kata

Tweet	Variabel Prediktor						
	x_1	…	x_{1158}	…	x_{1580}	…	x_{3689}
	Acara		Jatim		Lancar		Zona
1	0	…	0	…	1	…	0
2	0	…	1	…	0	…	0
3	0	…	0	…	0	…	0
⋮	⋮	…	⋮	…	⋮	…	⋮
1687	0	…	0	…	0	…	0

Berdasarkan hasil perhitungan praproses data dapat diketahui jumlah variabel prediktor/kata yang digunakan adalah 3689 kata. Dari semua kata yang digunakan, terdapat kata yang sering muncul. Berikut kata yang memiliki frekuensi kemunculan paling tinggi.

Tabel 4.5 Kata Frekuensi Tinggi

Kata	**Frekuensi**
Jalan	246
Macet	220
Mobil	127
Parkir	125
Polisi	112

Berdasarkan Tabel 4.5 dapat diketahui bahwa lima kata yang paling sering muncul pada semua data *tweet* masyarakat tentang Pemkot Surabaya adalah "Jalan" dengan frekuensi 246 kali muncul. Selanjutnya disusul oleh kata "Macet", "Mobil", "Parkir", dan pada urutan kelima adalah kata "Parkir". Selain data keseluruhan, data yang telah diklasifikasi juga telah dilakukan analisis praproses sehingga dapat diketahui kata-kata dengan

frekuensi tertinggi pada masing-masing klasifikasi. Berikut kata-kata yang memiliki frekuensi tertinggi pada klasifikasi positif dan negatif.

Tabel 4.6 Kata Frekuensi Tinggi Tiap Klasifikasi

Negatif		Positif	
Kata	**Frekuensi**	**Kata**	**Frekuensi**
Jalan	214	Polisi	37
Macet	186	Tugas	35
Parkir	116	Jalan	32
Mobil	111	Lancar	27
Lampu	96	Macet	22

Berdasarkan pada Tabel 4.6 dapat diketahui lima kata yang memiliki frekuensi tertinggi pada klasifikasi positif dan negatif. Pada klasifikasi positif kata yang paling sering muncul adalah "Polisi" dan pada urutan selanjutnya adalah "Tugas" dengan frekuensi masing-masing adalah 37 dan 35 Sedangkan pada klasifikasi negatif, kata "Jalan" adalah kata yang sering muncul pada *tweet* masyarakat tentang Pemkot Surabaya dengan frekuensi sebesar 214. Pada urutan kedua adalah kata "Macet" dan urutan selanjutnya adalah "Parkir", "Mobil" dan "Lampu". Berdasarkan daftar kata-kata yang sering muncul dapat diketahui bahwa kata-kata tersebut merupakan kata-kata yang berpengaruh dalam pembangunan model klasifikasi. selain dilakukan perhitungan kata yang sering muncul, dapat dilihat pula berdasarkan visual dengan menggunakan visualisasi *Word Cloud*.

4.3 Visualisasi *Word Cloud*

Visualisasi data dengan *Word Cloud* digunakan untuk mengetahui variabel prediktor (kata) yang sering digunakan atau sering muncul pada data *tweet*. Data yang digunakan pada *Word Cloud* ini adalah data *tweet* yang telah dibedakan berdasarkan klasifikasinya yaitu positif atau negatif. Sehingga output yang diperoleh ada dua *Word Cloud* yakni *Word Cloud* untuk data yang masuk dalam klasifikasi positif dan *Word Cloud* untuk data yang masuk dalam klasifikasi negatif.

Kata yang sering muncul akan memiliki ukuran lebih besar dari pada kata-kata lain yang jarang muncul, sebaliknya kata dengan frekuensi kemunculan yang kecil akan berukuran lebih kecil daripada kata yang lainnya. Tujuan ingin diketahui kata dengan frekuensi terbesar yaitu untuk mengetahui bidang apa yang sangat diapresiasi oleh masyarakat terhadap kinerja Pemkot Surabaya dan bidang apa yang menurut masyarakat masih perlu ditingkatkan oleh Pemkot Surabaya. Dilakukan analisis *Word Cloud* pada data yang diperoleh dari akun respi Pemkot Surabaya (@SapawargaSby), akun resmi Radio Suara Surabaya (@e100ss) dan gabungan antar kedua akun tersebut.

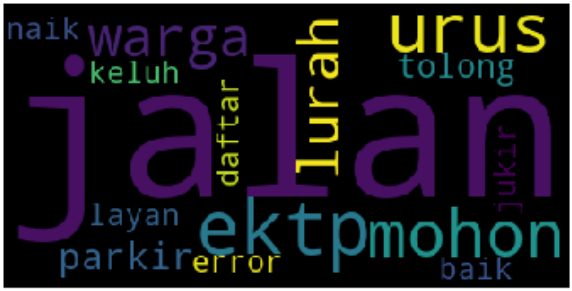
4.3.1 Visualisasi *Word Cloud* Akun Pemkot Surabaya

Data yang digunakan merupakan data dari akun resmi Pemkot Surabaya @SapawargaSby. Berdasarkan analisis yang telah dilakukan, hasil visualisasi *Word Cloud* adalah sebagaimana pada Gambar 4.4 dan Gambar 4.5 sebagai berikut.

Gambar 4.4 Visualisasi *Word Cloud* Pemkot Surabaya Sentimen Positif

Berdasarkan Gambar 4.4 dapat diketahui bahwa kata "Terimakasih" memiliki frekuensi dibicaran paling banyak. Kata selanjutnya yaitu "Warga", "Cepat", "Jelas", "Baik", "Informasi", "Tertib" dan kata-kata yang lainnya. Hal ini menunjukan bahwa masyarakat sangat mengapresiasi pada kinerja Pemkot dalam menjalankan tugasnya. Mayoritas *tweet* masyarakat kepada akun @SapawargaSby mengapresiasi kinerja Pemkot Surabaya dalam hal ketertiban dan penyampaian informasi.

Selanjutnya dengan menggunakan data dari akun @SapawargaSby, dilakukan visualisasi untuk data *tweet* yang masuk pada kategori negatif. Dari visualisasi menggunakan *Word Cloud* ini dilakukan untuk mengetahui keluhan masyarakat mengenai kinerja Pemkot Surabaya. Sehingga Pemkot Surabaya dapat melakukan peningkatan kinerja dalam hal tersebut. *Word Cloud* untuk data *tweet* kategori negatif dapat dilihat pada Gambar 4.5 berikut.



Gambar 4.5 Visualisasi *Word Cloud* Pemkot Surabaya Sentimen Negatif

Gambar 4.5 menunjukan bahwa kata "Jalan" memiliki frekuensi dibicaran paling banyak. Kata selanjutnya yaitu "e-KTP", "Mohon", "Urus", "Parkir, "Lurah" dan kata-kata lainnya. Hal ini menunjukan bahwa masyarakat masih mengeluhkan keadaan lalulintas di Surabaya dan layanan mengenai e-KTP. Parkir juga masih emnjadi perbincangan masyarakat dalam konteks negatif, sehingga dapat dijadikan informasi tambahan kepada Pemkot Surabaya mengenai bidang yang masih perlu diperbaiki dan menjadi sorotan negatif dari masyarakat terhadap Pemkot Surabaya yaitu mengenai lalulintas, e-KTP, dan parkir.

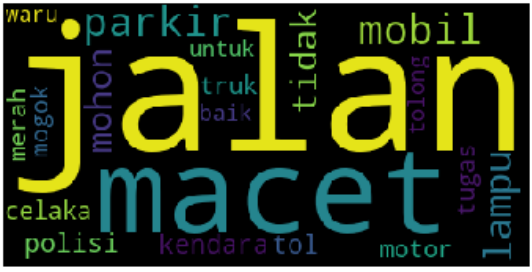
4.3.2 Visualisasi *Word Cloud* Akun Radio Suara Surabaya

Data selanjutnya yang digunakan adalah data dari akun resmi Radio Suara Surabaya @e100ss. Hasil dari analisis yang telah dilakukan sebagaimana pada Gambar 4.6 dan Gambar 4.7 sebagai berikut.

Gambar 4.6 Visualisasi *Word Cloud* e100ss Sentimen Positif

Berdasarkan hasil analisis pada Gambar 4.6 dapat diketahui bahwa kata "Polisi", "Tugas", "Lalin", "Jalan" dan "Jalan" merupakan kata yang lebih berukuran lebih besar dari pada kata yang lainnya. Hal ini menunjukan bahwa masyarakat memberikan apresiasi pada Pemkot melalui akun resmi Radio Suara Surabaya dalam menjalankan tugasnya, terutama pada Kepolisian. Kepolisian dalam menjalankan tugas sehingga tidak terjadi macet sangat diapresiasi oleh masyarakat.

Selanjutnya dengan menggunakan data dari akun resmi Radio Suara Surabaya @e100ss, dilakukan visualisasi pada kategori negatif. *Word Cloud* untuk data *tweet* kategori negatif dari akun @e100ss dapat dilihat pada Gambar 4.7 berikut.



Gambar 4.7 Visualisasi *Word Cloud* Sentimen Negatif

Gambar 4.7 menunjukan bahwa kata "Jalan" dan "Macet" memiliki frekuensi dibicaran paling banyak. Kata selanjutnya yaitu "Parkir", "Mogok", "Celaka", "Lampu", "Mobil", dan kata-

kata yang lainnya. Hal ini menunjukan bahwa masyarakat masih mengeluhkan kemacetan lalulintas, kecelakaan, dan parkir di Surabaya. Sehingga dapat dijadikan informasi tambahan kepada Pemkot Surabaya mengenai bidang yang masih perlu diperbaiki dan menjadi sorotan negatif dari masyarakat terhadap Pemkot Surabaya. hal-hal yang masih menjadi sorotan yaitu kemacetan di Surabaya yang dapat disebabkan olek kendaraan yang mogok atau kejadian-kejadian kecelakaan serta parkir juga dapat menjadi salah satu penyebabnya.

4.3.3 Visualisasi *Word Cloud* Akun Pemkot Surabaya dan Radio Suara Surabaya

Data yang digunakan merupakan data gabungan dari kedua akun yaitu @SapawargaSby dan @e100ss. Berdasarkan analisis yang telah dilakukan, hasil visualisasi *Word Cloud* dapat dilihat pada Gambar 4.8 dan Gambar 4.9 sebagai berikut.

Gambar 4.8 Visualisasi *Word Cloud* Data Positif

Berdasarkan Gambar 4.8 dapat diketahui bahwa kata “Polisi” memiliki frekuensi dibicaran paling banyak yang disusul dengan kata “Tugas”, “Jalan”, “Jadi”, “Lancar”, dan kata-kata yang lainnya. Hal ini menunjukan bahwa masyarakat sangat mengapresiasi pada pihak Kepolisian dalam menjalankan tugasnya. Mayoritas pada data *tweet* masyarakat mengapresiasi pihak polisi lalu lintas dalam pengaturan jalan. Apresiasi lain juga ditujukan kepada Dinas Perhubungan mengenai pemasangan CCTV pada hampir semua *traffic light* di Surabaya.

Selanjutnya dengan menggunakan data gabungan dari akun resmi Pemkot Surabaya dan Radio Suara Surabaya, dilakukan visualisasi untuk data *tweet* yang masuk pada kategori negatif. Dari visualisasi menggunakan *Word Cloud* ini dilakukan untuk mengetahui keluhan masyarakat mengenai Pemkot Surabaya. Sehingga Pemkot Surabaya dapat melakukan peningkatan dalam hal tersebut. *Word Cloud* untuk data *tweet* kategori negatif dapat dilihat pada Gambar 4.9 berikut.

Gambar 4.9 Visualisasi *Word Cloud* Data Negatif

Gambar 4.9 menunjukan bahwa kata “Jalan” dan “Macet” sangat dominan, yang mengindikasikan bahwa masyarakat masih mengeluhkan keadaan lalulintas di Surabaya tentang jalan yang macet. Kemacetan di Surabaya sangat banyak dibincangkan oleh masyarakat terhadap Pemkot Surabaya, sehingga dapat dijadikan informasi tambahan kepada Pemkot Surabaya mengenai bidang yang masih perlu diperbaiki dan menjadi sorotan negatif dari masyarakat terhadap pihak Pemkot Surabaya.

4.4 Metode Klasifikasi *Naïve Bayes Classifier* (NBC)

Sejalan dengan tujuan penelitian ini adalah mendapatkan hasil kasifikasi sentimen terbaik maka perlu dilakukan pemilikan metode klasifikasi yang terbaik pula. Metode klasifikasi yang digunakan adalah *Naïve Bayes Classifier*, *Support Vector Machine* dan Regresi Logistik Biner sebagai perbandingan metode klasiknya. Pemilihan metode terbaik didasarkan pada

rata-rata *Area Under Curve* (AUC), *accuracy*, *precision*, dan *recall*. Rata-rata diperoleh dari masing-masing *subset fold* dengan meode pembagian data *cross validation* 10 *fold*. Berdasarkan metode pembagian data *training* dan *testing* dengan metode *cross validation*, data *tweet* sebanyak 1686 dibagi menjadi 90% data *training* dan 10% data *testing* sebanyak 10 macam kombinasi (*subset fold*). Sehingga data *training* pada setiap *subset* adalah 1517 *tweet* dan data *training* sebanyak 169 *tweet*.

Sebagaimana telah dipaparkan pada subbab karakteristik data, dapat diketahui bahwa jumlah *tweet* yang mengandung sentimen negatif lebih banyak daripada positif. Prosentase perbandingan antara jumlah *tweet* yang mengandung sentimen negatif dan positif adalah 78:22 dimana angka tersebut menunjukan bahwa data imbalance. Oleh karena itu data *tweet* akan dilakukan *oversampling* Synthetic Minority *Oversampling* TEchnique (SMOTE), dimana jumlah data yang mengandung sentimen positif akan disamakan dengan jumlah data yang mengandung sentimen negatif. Sehingga untuk setiap pengamatan prosentase perbandingan positif dan negatif adalah 50:50. Dari seluruh jumlah data sebanyak 1686 *tweet* terdapat 1318 *tweet* yang mengandung sentimen negatif dan 369 yang mengandung sentimen positif, akan tetapi setelah dilakukan SMOTE jumlah *tweet* yang mengandung sentimen positif dan negatif adalah sama yaitu 1318 dengan jumlah total *tweet* menjadi 2636 *tweet*.

Berdasarkan data awal dan data hasil SMOTE tersebut dilakukan perhitungan sehingga diperoleh model dan prediksi sehingga dapat diketahui performa klasifikasi dengan menggunakan metode *Naïve Bayes Classifier* sebagai berikut.

4.4.1 Metode *Naïve Bayes Classifier* Data Awal

Metode klasifikasi *Naïve Bayes Classifier* (NBC) merupakan salah satu metode yang populer dalam analisis klasifikasi. Metode ini sering digunakan karena mudah dan cepat dalam pengerjaannya. Klasifikasi menggunakan metode NBC menghasilkan sebuah model, dimana model tersebut dapat digunakan untuk mengklasifikasikan data-data selanjutnya. Telah

dilakukan perhitungan model sebagaimana pada Lampiran 8 dimana data yang digunakan untuk membuat model adalah data *training* dan akan menerapkan model tersebut kepada data *testing*. Sebagaimana telah dijelaskan sebelumnya bahwa pembagian data *training* dan *testing* dilakukan dengan menggunakan *cross validation* 10-*fold*s, sehingga dapat diperoleh hasil setiap *subset* 10 *fold* adalah sebagai berikut.

Tabel 4.7 Pemilihan S*ubset* Tebaik NBC Data Awal

Subset 10-Folds	***Accuracy***	***Precision***	***Recall***	**AUC**
Subset ke-1	0.74	0.18	0.05	0.49
Subset ke-2	0.78	0.46	0.16	0.55
Subset ke-3	0.80	0.67	0.16	0.57
Subset ke-4	0.77	0.38	0.08	0.52
Subset ke-5	0.77	0.38	0.08	0.52
Subset ke-6	0.72	0.26	0.16	0.52
Subset ke-7	0.80	0.70	0.19	0.58
Subset ke-8	0.79	0.67	0.05	0.52
Subset ke-9	0.75	0.00	0.00	0.48
Subset ke-10	0.80	0.62	0.14	0.56
Rata-rata	0.77	0.43	0.11	0.53

Sebagaimana pada Tabel 4.7 dapat pada hasil nilai *accuracy*, *precision*, *recall* dan AUC dengan metode NBC pada data awal. *Subset* ke-7 memiliki nilai performa klasifikasi yang terbaik yaitu nilai *accuracy* sebesar 80%, *precision* sebesar 70%, *recall* sebesar 19% dan nilai AUC sebesar 0,58. Nilai *precision* dan *recall* disini sangat kecil karena dapat diketahui bahwa *tweet* yang mengandung sentimen positif adalah minoritas, dimana perhitungan *precision* dan *recall* menghitung ketepatan untuk data positif. Berdasarkan pada Tabel 4.7 tersebut, *subset* ke-3 yang terbaik dan dengan kombinasi data *training* dan *testing* data pada *subset* ini akan dilakukan permodelan sehingga pada data selanjutnya dapat langsung menggunakan model tersebut. Model untuk metode NBC berdasarkan pada *subset* ke-3 adalah sebagai berikut.

Tabel 4.8 Model Klasifikasi NBC Data Awal

Klasifikasi	Model
Negatif	$0,78 \times 0,00022^{(f1)} \times \ldots \times 0,00050^{(f3687)} \times 0,00036^{(f3688)}$
Positif	$0,22 \times 0,00050^{(f1)} \times \ldots \times 0,00116^{(f3687)} \times 0,00083^{(f3688)}$

Berdasarkan model klasifikasi dengan metode NBC pada Tabel 4.8 dapat dilakukan pengklasifikasian pada data selanjutnya. Pengklasifikasian dilakukan dengan memasukkan frekuensi kata pada data baru pada (f_1) sampai dengan (f_{3688}) disetiap masing masing klasifikasi positif dan negatif. Sebagai contoh penerapannya akan dilakukan pada data *subset* ke-7 juga untuk mengetahui hasil prediksi untuk masing-masing *tweet* dan *confussion matrix* yang diperoleh.

Tabel 4.9 Probabilitas Klasifikasi NBC Data Awal

Tweet	Probabilitas Negatif	Probabilitas Positif	Keputusan
1	0.999	0.001	Negatif
2	0.183	0.817	Positif
⋮	⋮	⋮	⋮
85	0.998	0.002	Negatif
⋮	⋮	⋮	⋮
168	0.020	0.980	Positif
169	0.967	0.033	Negatif

Berdasarkan pada Tabel 4.9 yang telah diperoleh, maka dapat diketahui peluang positif dan negatif sehingga dapat diputuskan bahwa suatu *tweet* masuk pada klasifikasi positif atau negatif. data *tweet* pertama memiliki peluang masuk pada klasifikasi negatif sebesar 0,999 sedangkan peluang masuk kepada klasifikasi positif sebesar 0,001 sehingga dapat diputuskan bahwa data *tweet* pertama tersebut masuk dalam klasifikasi positif. Dengan cara yang sama juga dilakukan pada *tweet* kedua hingga *tweet* pada data *testing* terakhir yaitu 169 yang memiliki peluang negatif 0,967 dan peluang positif sebesar 0,033. Sehingga data ke-169 masuk pada klasifikasi negatif. *Output* selanjutnya yang dihasilkan adalah *confusion matrix* guna

melakukan perhitungan ketepatan klasifikasi. *Confusion matrix* yang diperoleh akan dihitung nilai *accuracy*, *precison*, *recall*, dan AUC dari metode NBC yang ditampilkan pada Tabel 4.10 dan berikut.

Tabel 4.10 *Confusion matrix* NBC Data Awal

Kelas Aktual	**Kelas Prediksi**	
	Negatif	Positif
Negatif	129	3
Positif	30	7

$$accuracy = \frac{129+7}{129+3+30+7} = 0.80$$

$$precision = \frac{7}{7+3} = 0.70$$

$$recall = \frac{3}{3+30} = 0.19$$

$$AUC = \frac{1}{2}\left(\frac{129}{129+3} + \frac{7}{7+30}\right)$$

$$= 0.58$$

Berdasarkan Tabel *confusion matrix* diatas dapat diketahui bahwa ketepatan klasifikasi untuk data negatif yang diklasifikasikan benar negatif adalah 129, sedangkan untuk data negatif diklasifikasikan positif adalah 3. Data positif yang diklasifikasikan negatif sebesar 30, dan untuk data *tweet* negatif diklasifikasikan benar negatif adalah 7. Berdasarkan pada data dari *confusion matrix* yang telah diperoleh, dilakukan perhitungan untuk mengetahui nilai AUC. Setelah dilakukan perhitungan maka dapat diperoleh nilai AUC 0,58 dengan menggunakan metode NBC.

Sebagaimana telah dijabarkan sebelumnya bahwa data yang digunakan merupakan data imbalance sehingga nilai perfirma klasifikasi yang diperoleh masih tergolong pada kategori kurang. Selanjutnya dilakukan analisis klasifikasi dengan metode *Naïve Bayes Classifier* dengan menggunakan yang telah dilakukan *oversampling* SMOTE.

4.4.2 Metode *Naïve Bayes Classifier* Data SMOTE

Metode *oversampling* SMOTE merupakan metode balancing data dengan cara menyamakan jumlah data pada minoritas (data bersentimen positif) dengan data mayoritas (data bersentimen negatif). Penyamaan jumlah minoritas dengan jumlah mayoritas tersebut dilakukan dengan cara replika secara acak. Telah dilakukan analisis *Naïve Bayes Classifier* dengan data hasil *oversampling* SMOTE menggunakan *cross validation* 10 *fold* sehingga diperoleh performa klasifikasi untuk masing-masing *subset fold*nya adalah sebagai berikut.

Tabel 4.11 Pemilihan S*ubset* Tebaik NBC data SMOTE

***Subset* 10-Folds**	***Accuracy***	***Precision***	***Recall***	**AUC**
Subset ke-1	0.75	0.45	0.59	0.65
Subset ke-2	0.75	0.45	0.62	0.70
Subset ke-3	0.80	0.55	0.59	0.73
Subset ke-4	0.76	0.47	0.70	0.74
Subset ke-5	0.74	0.40	0.38	0.61
Subset ke-6	0.56	0.27	0.62	0.58
Subset ke-7	0.74	0.43	0.59	0.69
Subset ke-8	0.72	0.41	0.65	0.70
Subset ke-9	0.80	0.56	0.41	0.66
Subset ke-10	0.77	0.47	0.64	0.72
Rata-rata	0.74	0.45	0.58	0.68

Sebagaimana pada Tabel 4.11 dapat diketahui hasil nilai *accuracy*, *precision*, *recall* dan AUC dengan metode NBC pada data hasil *oversampling* SMOTE. Dimana pada *subset* ke-7 memiliki nilai performa klasifikasi yang terbaik yaitu nilai *accuracy* sebesar 76%, *precision* sebesar 47%, *recall* sebesar 70% dan nilai AUC sebesar 0,74. Berdasarkan pada Tabel 4.10 tersebut, *subset* ke-4 merupakan *subset* yang terbaik dan dengan kombinasi data *training* dan data *testing* pada *subset* ini akan dilakukan permodelan sehingga pada data selanjutnya dapat langsung menggunakan model tersebut. Model untuk metode NBC berdasarkan pada *subset* ke-7 data hasil *oversampling* SMOTE adalah sebagai berikut.

Tabel 4.12 Model Klasifikasi NBC data SMOTE

Klasifikasi	Model
Positif	$0,50 \times 0,00045^{(f1)} \times \ldots \times 0,00075^{(f3687)} \times 0,00064^{(f3688)}$
Negatif	$0,50 \times 0,00030^{(f1)} \times \ldots \times 0,00051^{(f3687)} \times 0,00043^{(f3688)}$

Berdasarkan model klasifikasi dengan metode NBC pada Tabel 4.8 dapat dilakukan pengkasifikasian pada data selanjutnya. Pengklasifikasian dilakukan dengan memasukkan frekuensi kata pada data baru pada (f_1) sampai dengan (f_{3688}) disetiap masing masing klasifikasi positif dan negatif. Sebagai contoh penerapannya dakan dilakukan pada data *subset* terbaik yaitu *subset* ke-10 untuk mengetahui hasil prediksi untuk masing-masing *tweet* dan *confussion matrix* yang diperoleh.

Tabel 4.13 Probabilitas Klasifikasi NBC data SMOTE

Tweet	Probabilitas Negatif	Probabilitas Positif	Keputusan
1	0.023	0.977	Positif
2	0.056	0.943	Positif
⋮	⋮	⋮	⋮
85	0.998	0.002	Negatif
⋮	⋮	⋮	⋮
168	0.574	0.426	Negatif
169	0.316	0.684	Positif

Berdasarkan pada Tabel 4.13 yang diperoleh, maka dapat diketahui peluang positif dan negatif sehingga dapat diputuskan bahwa suatu *tweet* masuk pada klasifikasi positif atau negatif. Data *tweet* pertama memiliki peluang masuk pada klasifikasi negatif sebesar 0,023 sedangkan peluang masuk kepada klasifikasi positif sebesar 0,977 sehingga dapat diputuskan bahwa data *tweet* pertama tersebut masuk dalam klasifikasi positif. *Output* selanjutnya yang dihasilkan adalah *confusion matrix* guna melakukan perhitungan ketepatan klasifikasi. *Confusion matrix* yang diperoleh akan dihitung nilai *accuracy*, *precison, recall,* dan AUC dari metode NBC yang ditampilkan pada Tabel 4.12 berikut.

Tabel 4.14 *Confusion matrix* NBC Data SMOTE

Kelas Aktual	Kelas Prediksi	
	Negatif	Positif
Negatif	103	29
Positif	11	26

$$accuracy = \frac{103+26}{103+26+11+29} = 0,76$$

$$precision = \frac{26}{26+29} = 0,47$$

$$recall = \frac{26}{26+11} = 0,70$$

$$AUC = \frac{1}{2}\left(\frac{103}{103+26} + \frac{26}{26+11}\right)$$

$$= 0,74$$

Berdasarkan Tabel *confusion matrix* diatas dapat diketahui bahwa ketepatan klasifikasi untuk data negatif yang diklasifikasikan benar negatif adalah 103, sedangkan untuk data negatif diklasifikasikan positif adalah 29. Data positif yang diklasifikasikan negatif sebesar 11, dan untuk data *tweet* negatif diklasifikasikan benar negatif adalah 26. Berdasarkan pada data dari *confusion matrix* yang telah diperoleh, dilakukan perhitungan untuk mengetahui nilai AUC. Setelah dilakukan perhitungan maka dapat diperoleh nilai AUC 0,57 dengan menggunakan metode NBC pada data hasil *oversampling* SMOTE.

Selanjutnya akan dilakukan perbandingan antara performa klasifikasi data awal dengan data yang telah dilakukan *oversampling* SMOTE pada metode NBC. Perbandingan performa klasifikasi dilakukan dengan membandingkan rata-rata performa klasifikasi setiap *subset fold*, sehingga diperoleh ringakasan sebagai berikut.

Tabel 4.15 Perbandingan Performa Klasifikasi NBC Data Awal dan SMOTE

Data	*Accuracy*	*Precision*	*Recall*	AUC
Data Awal	0.77	0.43	0.11	0.53
Data SMOTE	0.74	0.45	0.58	0.68

Berdasarkan pada Tabel 4.15 dapat diketahui bahwa data hasil *oversampling* SMOTE menghasilkan rata-rata performa klasifikasi lebih baik dari pada data awal.

4.5 Metode Klasifikasi *Support Vector Machine* (SVM)

Metode klasifikasi *Support Vector Machine* (SVM) merupakan salah satu metode yang banyak digunakan untuk analisis klasifikasi. Metode ini sering digunakan karena ketepatan klasifikasi yang dihasilkan cenderung tinggi. Pada metode ini terdapat beberapa *kernel* yang dapat menunjang analisis klasifikasi, akan tetapi pada penelitian ini tidak semua *kernel* akan digunakan. *Kernel* yang digunakan adalah *kernel Linear* dan *Radial Basis Function* (RBF). Analisis klasifikasi menggunakan metode SVM dengan *kernel Linear*, dimana pada analisis menggunakan *kernel* ini membutuhkan parameter *cost* (*C*) yang akan dilakukan pemilihan nilai parameter yang optimum. Nilai parameter *C* akan dilakukan uji coba mulai dari 10^{-2} sampai dengan 10^{2}. Sedangkan analisis menggunakan klasifikasi SVM *kernel* RBF menggunakan parameter *C* dan γ dimana nilai parameter yang akan dicobakan sama yaitu mulai dari 10^{-2} sampai dengan 10^{2}. Sebagaimana pada metode sebelumnya yaitu metode NBC, data yang digunakan adalah data awal dengan jumlah data sebanyak 1686 *tweet* terdapat 1318 *tweet* yang mengandung sentimen negatif dan 369 yang mengandung sentimen positif. juga akan dilakukan dengan menggunakan data hasil *oversampling* SMOTE dengan jumlah *tweet* yang mengandung sentimen positif dan negatif adalah sama yaitu 1318 dengan jumlah total *tweet* menjadi 2636 *tweet*.

Berdasarkan data awal dan data hasil *oversampling* SMOTE tersebut dilakukan perhitungan sehingga diperoleh hasil performa klasifikasi sehingga diketahui nilai parameter *C* yang optimum, serta dapat diketahui *confusion matrix* dari hasil analisis klasifikasi dengan menggunakan nilai parameter sebagai berikut.

4.5.1 Metode *Suppor Vector Machine* (SVM) *Kernel Linear* pada Data Awal

Klasifikasi menggunakan metode *Support Vector Machine* (SVM) *kernel Linear* akan menentukan nilai parameter *C* yang optimum. Untuk mengetahui nilai parameter *C* yang optimum, dilakukan perhitungan performa klasifikasi pada masing-masing nilai parameter yang dicobankan, sehingga nilai parameter yang hasilkan performa klasifikasi paling tinggi adalah nilai parameter yang paling optimum. Sebagaimana pada metode sebelumnya, Pembagian data *training* dan *testing* dilakukan dengan menggunakan *cross validation* 10-*fold*s, sehingga dapat diperoleh hasil setiap *subset* 10 *fold* pada masing-masing nilai parameter *C* adalah sebagai berikut.

Tabel 4.16 Pemilihan *Subset* Tebaik SVM *kernel Linear* Data Awal

Nilai Parameter (C)	***Subset 10-Folds***	***Accuracy***	***Precision***	***Recall***	**AUC**
10^{-2}	*Subset* ke-1	0.78	0.00	0.00	0.50
	Subset ke-2	0.78	0.00	0.00	0.50
	Subset ke-3	0.78	0.00	0.00	0.50
	Subset ke-4	0.78	0.00	0.00	0.50
	Subset ke-5	0.78	0.00	0.00	0.50
	Subset ke-6	0.78	0.00	0.00	0.50
	Subset ke-7	0.78	0.00	0.00	0.50
	Subset ke-8	0.78	0.00	0.00	0.50
	Subset ke-9	0.78	0.00	0.00	0.50
	Subset ke-10	0.78	0.00	0.00	0.50
Rata-rata		0.78	0.00	0.00	0.50
10^{-1}	*Subset* ke-1	0.78	0.00	0.00	0.50
	Subset ke-2	0.78	0.00	0.00	0.50
	Subset ke-3	0.78	0.00	0.00	0.50
	Subset ke-4	0.78	0.00	0.00	0.50
	Subset ke-5	0.78	0.00	0.00	0.50
	Subset ke-6	0.78	0.00	0.00	0.50
	Subset ke-7	0.78	0.00	0.00	0.50
	Subset ke-8	0.78	0.00	0.00	0.50
	Subset ke-9	0.78	0.00	0.00	0.50
	Subset ke-10	0.78	0.00	0.00	0.50
Rata-rata		0.78	0.00	0.00	0.50

Tabel 4.16 Pemilihan *Subset* Tebaik SVM *kernel Linear* Data Awal *(Lanjutan)*

Nilai Parameter (C)	*Subset 10-Folds*	Accuracy	Precision	Recall	AUC
10^0	*Subset* ke-1	0.81	0.73	0.22	0.60
	Subset ke-2	0.82	0.73	0.30	0.63
	Subset ke-3	0.84	0.86	0.32	0.65
	Subset ke-4	0.82	0.88	0.19	0.59
	Subset ke-5	0.79	0.75	0.08	0.54
	Subset ke-6	0.71	0.32	0.30	0.56
	Subset ke-7	0.80	0.75	0.16	0.57
	Subset ke-8	0.82	0.80	0.22	0.60
	Subset ke-9	0.83	0.90	0.24	0.62
	Subset ke-10	0.84	0.74	0.39	0.68
Rata-rata		0.81	0.75	0.24	0.61
10^1	*Subset* ke-1	0.77	0.46	0.32	0.61
	Subset ke-2	0.80	0.55	0.43	0.67
	Subset ke-3	0.79	0.54	0.38	0.64
	Subset ke-4	0.79	0.53	0.46	0.67
	Subset ke-5	0.78	0.45	0.14	0.54
	Subset ke-6	0.67	0.32	0.43	0.59
	Subset ke-7	0.79	0.56	0.27	0.60
	Subset ke-8	0.82	0.63	0.46	0.69
	Subset ke-9	0.76	0.73	0.76	0.58
	Subset ke-10	0.79	0.51	0.50	0.69
Rata-rata		0.78	0.53	0.42	0.63
10^2	*Subset* ke-1	0.73	0.41	0.49	0.64
	Subset ke-2	0.73	0.42	0.59	0.68
	Subset ke-3	0.74	0.43	0.59	0.69
	Subset ke-4	0.80	0.39	0.65	0.68
	Subset ke-5	0.76	0.76	0.76	0.64
	Subset ke-6	0.54	0.71	0.54	0.68
	Subset ke-7	0.74	0.41	0.46	0.64
	Subset ke-8	0.72	0.73	0.72	0.61
	Subset ke-9	0.74	0.72	0.74	0.58
	Subset ke-10	0.78	0.78	0.78	0.68
Rata-rata		0.73	0.58	0.63	0.65

Sebagaimana pada Tabel 4.16 dapat diketahui hasil nilai *accuracy*, *precision*, *recall* dan AUC dengan metode SVM *kernel Linear* pada data awal, dimana pada nilai parameter 10^2 di *subset* ke-3 memiliki nilai performa klasifikasi yang terbaik yaitu nilai *accuracy* sebesar 74%, *precision* sebesar 43%, *recall* sebesar

59% dan nilai AUC sebesar 0,69. Berdasarkan pada Tabel 4.16 tersebut, *subset* ke-3 merupakan *subset* yang terbaik dengan kombinasi data *training* dan data *testing* yang menghasilkan *confusion matrix* sebagai berikut.

Tabel 4.17 *Confusion matrix* SVM *Linear* Data Awal

Kelas Aktual	Kelas Prediksi	
	Negatif	Positif
Negatif	103	29
Positif	15	22

Berdasarkan tabel *confusion matrix* diatas dapat diketahui bahwa ketepatan klasifikasi untuk data negatif yang diklasifikasikan benar negatif adalah 103, sedangkan untuk data negatif diklasifikasikan positif adalah 29. Data positif yang diklasifikasikan negatif sebesar 15, dan untuk data *tweet* negatif diklasifikasikan benar negatif adalah 22. Sehingga dapat putuskan bahwa nilai parameter *C* optimum pada kasus klasifikasi pendapat masyarakat terhadap Pemkot Surabaya dengan metode SVM *kernel Linear* adalah 10 dengan perolehan nilai AUC sebesar 0,69.

4.5.2 Metode *Suppor Vector Machine* (SVM) *Kernel Linear* pada Data SMOTE

Selanjutnya dilakukan analisis kasifikasi menggunakan metode *Support Vector Machine* (SVM) dengan *kernel Linear* pada data hasil *oversampling* SMOTE. Sebagaimana pada data awal, pembagian data *training* dan *testing* dilakukan dengan menggunakan *cross validation* 10-*fold*s, sehingga dapat diperoleh hasil setiap *subset* 10 *fold* pada masing-masing nilai parameter *C* adalah sebagai berikut.

Tabel 4.18 Pemilihan *Subset* Tebaik SVM *kernel Linear* Data SMOTE

Nilai Parameter (C)	*Subset 10-Folds*	*Accuracy*	*Precision*	*Recall*	AUC
10^{-2}	*Subset* ke-1	0.73	0.41	0.54	0.66
	Subset ke-2	0.59	0.30	0.65	0.61
	Subset ke-3	0.62	0.34	0.78	0.68
	Subset ke-4	0.59	0.31	0.73	0.64
	Subset ke-5	0.54	0.24	0.51	0.53

	Subset ke-6	0.76	0.44	0.32	0.61
	Subset ke-7	0.83	0.66	0.51	0.72
	Subset ke-8	0.64	0.35	0.76	0.69
	Subset ke-9	0.80	0.57	0.32	0.63
	Subset ke-10	0.80	0.55	0.50	0.69
Rata-rata		0.69	0.42	0.56	0.65
10^{-1}	*Subset* ke-1	0.62	0.34	0.81	0.69
	Subset ke-2	0.78	0.33	0.70	0.65
	Subset ke-3	0.62	0.34	0.81	0.69
	Subset ke-4	0.58	0.31	0.73	0.63
	Subset ke-5	0.53	0.23	0.51	0.50
	Subset ke-6	0.43	0.24	0.24	0.54
	Subset ke-7	0.75	0.45	0.76	0.75
	Subset ke-8	0.78	0.33	0.73	0.66
	Subset ke-9	0.80	0.55	0.65	0.75
	Subset ke-10	0.75	0.46	0.86	0.79
Rata-rata		0.66	0.36	0.68	0.67
10^{0}	*Subset* ke-1	0.73	0.42	0.54	0.66
	Subset ke-2	0.72	0.40	0.62	0.68
	Subset ke-3	0.75	0.45	0.76	0.75
	Subset ke-4	0.71	0.41	0.78	0.74
	Subset ke-5	0.69	0.33	0.38	0.58
	Subset ke-6	0.54	0.27	0.65	0.58
	Subset ke-7	0.72	0.39	0.51	0.64
	Subset ke-8	0.71	0.38	0.54	0.65
	Subset ke-9	0.81	0.56	0.68	0.76
	Subset ke-10	0.75	0.46	0.89	0.80
Rata-rata		0.71	0.41	0.64	0.68
10^{1}	*Subset* ke-1	0.73	0.42	0.57	0.67
	Subset ke-2	0.69	0.37	0.57	0.65
	Subset ke-3	0.72	0.42	0.68	0.71
	Subset ke-4	0.69	0.39	0.76	0.71
	Subset ke-5	0.74	0.41	0.41	0.62
	Subset ke-6	0.54	0.27	0.65	0.58
	Subset ke-7	0.69	0.35	0.49	0.61
	Subset ke-8	0.73	0.40	0.51	0.65
	Subset ke-9	0.73	0.42	0.62	0.69
	Subset ke-10	0.79	0.51	0.94	0.85
Rata-rata		*0.71*	0.41	0.64	0.68

Tabel 4.18 Pemilihan *Subset* Tebaik SVM *Linear* Data SMOTE *(Lanjutan)*

Nilai Parameter (C)	*Subset 10-Folds*	Accuracy	Precision	Recall	AUC
10^2	*Subset* ke-1	0.74	0.42	0.57	0.68
	Subset ke-2	0.73	0.41	0.65	0.70
	Subset ke-3	0.74	0.42	0.65	0.70
	Subset ke-4	0.68	0.38	0.76	0.71
	Subset ke-5	0.73	0.40	0.43	0.63
	Subset ke-6	0.54	0.27	0.65	0.58
	Subset ke-7	0.70	0.35	0.46	0.61
	Subset ke-8	0.69	0.35	0.49	0.62
	Subset ke-9	0.72	0.44	0.65	0.71
	Subset ke-10	0.77	0.48	0.92	0.82
Rata-rata		0.70	0.39	0.62	0.68

Sebagaimana pada Tabel 4.18 dapat diketahui hasil nilai *accuracy*, *precision*, *recall* dan AUC dengan metode SVM *kernel Linear* pada data hasil *oversampling* SMOTE, dimana pada nilai parameter 10^2 di *subset* ke-10 memiliki nilai performa klasifikasi yang terbaik yaitu nilai *accuracy* sebesar 75%, *precision* sebesar 46%, *recall* sebesar 89% dan nilai AUC sebesar 0,80. Berdasarkan pada Tabel 4.18 tersebut, *subset* ke-10 merupakan *subset* yang terbaik dengan kombinasi data *training* dan data *testing* yang menghasilkan *confusion matrix* sebagai berikut.

Tabel 4.19 Confusion matrix SVM *Linear* Data SMOTE

Kelas Aktual	Kelas Prediksi	
	Negatif	Positif
Negatif	94	37
Positif	4	32

Berdasarkan tabel *confusion matrix* diatas dapat diketahui bahwa ketepatan klasifikasi untuk data negatif yang diklasifikasikan benar negatif adalah 94, sedangkan untuk data negatif diklasifikasikan positif adalah 37. Data positif yang diklasifikasikan negatif sebesar 4, dan untuk data *tweet* negatif diklasifikasikan benar negatif adalah 32. Sehingga dapat putuskan bahwa nilai parameter *C* optimum pada kasus klasifikasi pendapat masyarakat terhadap Pemkot Surabaya dengan metode SVM *kernel Linear* adalah 1 dengan perolehan nilai AUC sebesar 0,80.

Selanjutnya akan dilakukan perbandingan antara performa klasifikasi data awal dengan data yang telah dilakukan *oversampling* SMOTE pada metode SVM *kernel Linear*. Perbandingan performa klasifikasi dilakukan dengan cara membandingkan rata-rata performa klasifikasi setiap *subset fold* pada nilai parameter terbaik yaitu 10^2, sehingga diperoleh ringkasan sebagai berikut.

Tabel 4.15 Perbandingan Performa Klasifikasi SVM *Linear*

Data	*Accuracy*	*Precision*	*Recall*	AUC
Data Awal	0.73	0.58	0.63	0.65
Data SMOTE	0.71	0.41	0.64	0.68

Berdasarkan pada Tabel 4.15 dapat diketahui bahwa data hasil *oversampling* SMOTE menghasilkan rata-rata performa klasifikasi lebih baik dari pada data awal.

4.5.3 Metode *Suppor Vector Machine* (SVM) *Kernel* RBF pada Data Awal

Klasifikasi menggunakan metode *Support Vector Machine* (SVM) *kernel* RBF akan menentukan nilai parameter *C* dan γ yang optimum. Untuk mengetahui nilai parameter *C* dan γ yang optimum, dilakukan perhitungan performa klasifikasi pada masing-masing nilai parameter yang dicobakan, sehingga nilai parameter yang hasilkan performa klasifikasi paling tinggi adalah nilai parameter yang paling optimum. Telah dilakukan analisis klasifikasi SVM *kernel* RBF pada data awal sebagaimana pada Lampiran 9, berikut merupakan hasil yang diperoleh dari setiap *subset* 10 *fold* pada masing-masing nilai parameter γ dan pada parameter $C=10^2$. Dipilih parameter $C=10^2$ yang ditampilkan, sebab pada parameter ini terdapat nilai optimum dari performa klasifikasinya.

Tabel 4.20 Pemilihan S*ubset* Tebaik SVM *kernel* RBF data Awal

Parameter γ	***Subset* 10-*Folds***	***Accuracy***	***Precision***	***Recall***	**AUC**
10^{-2}	*Subset* ke-1	0.79	0.56	0.27	0.60
	Subset ke-2	0.83	0.58	0.41	0.68
	Subset ke-3	0.83	0.76	0.35	0.66
	Subset ke-4	0.81	0.60	0.41	0.66
	Subset ke-5	0.79	0.55	0.16	0.56
	Subset ke-6	0.80	0.34	0.41	0.59
	Subset ke-7	0.80	0.61	0.30	0.62
	Subset ke-8	0.81	0.62	0.35	0.65
	Subset ke-9	0.82	0.77	0.27	0.62
	Subset ke-10	0.82	0.60	0.50	0.70
Rata-rata		0.81	0.60	0.34	0.63
10^{-1}	*Subset* ke-1	0.79	0.52	0.32	0.62
	Subset ke-2	0.80	0.57	0.32	0.65
	Subset ke-3	0.81	0.64	0.38	0.66
	Subset ke-4	0.79	0.48	0.35	0.67
	Subset ke-5	0.78	0.45	0.14	0.55
	Subset ke-6	0.67	0.33	0.41	0.58
	Subset ke-7	0.79	0.43	0.16	0.60
	Subset ke-8	0.84	0.71	0.32	0.70
	Subset ke-9	0.79	0.56	0.38	0.64
	Subset ke-10	0.81	0.53	0.47	0.72
Rata-rata		0.79	0.52	0.33	0.64
10^{0}	*Subset* ke-1	0.80	0.60	0.08	0.56
	Subset ke-2	0.81	1.00	0.11	0.58
	Subset ke-3	0.81	1.00	0.22	0.60
	Subset ke-4	0.81	1.00	0.16	0.58
	Subset ke-5	0.80	1.00	0.05	0.54
	Subset ke-6	0.74	0.32	0.19	0.57
	Subset ke-7	0.80	1.00	0.05	0.56
	Subset ke-8	0.81	1.00	0.16	0.58
	Subset ke-9	0.85	1.00	0.32	0.66
	Subset ke-10	0.84	0.86	0.33	0.69
Rata-rata		0.81	0.88	0.17	0.59

Tabel 4.20 Pemilihan S*ubset* Tebaik SVM *kernel* RBF data Awal *(Lanjutan)*

Parameter γ	***Subset* 10-*Folds***	***Accuracy***	***Precision***	***Recall***	**AUC**
10^{-2}	*Subset* ke-1	0.80	0.60	0.08	0.57
	Subset ke-2	0.81	1.00	0.11	0.58
	Subset ke-3	0.82	1.00	0.22	0.62
	Subset ke-4	0.81	1.00	0.16	0.58
	Subset ke-5	0.80	1.00	0.05	0.54
	Subset ke-6	0.74	0.32	0.19	0.57
	Subset ke-7	0.80	1.00	0.05	0.58
	Subset ke-8	0.81	1.00	0.16	0.58
	Subset ke-9	0.85	1.00	0.32	0.66
	Subset ke-10	0.84	0.86	0.33	0.69
Rata-rata		0.81	0.88	0.17	0.60
10^{-1}	*Subset* ke-1	0.78	0.00	0.00	0.50
	Subset ke-2	0.78	0.00	0.00	0.50
	Subset ke-3	0.78	0.00	0.00	0.50
	Subset ke-4	0.78	0.00	0.00	0.50
	Subset ke-5	0.79	1.00	0.03	0.51
	Subset ke-6	0.72	0.00	0.00	0.46
	Subset ke-7	0.78	0.00	0.00	0.50
	Subset ke-8	0.78	0.00	0.00	0.50
	Subset ke-9	0.80	1.00	0.11	0.55
	Subset ke-10	0.81	1.00	0.11	0.56
Rata-rata		0.78	0.30	0.03	051

Sebagaimana pada Tabel 4.20 dapat diketahui hasil nilai *accuracy*, *precision*, *recall* dan AUC dengan metode SVM *kernel* RBF pada data awal, dimana pada nilai parameter $C=10^2$ dan $\gamma=10^{-1}$ di*subset* ke-10 memiliki nilai performa klasifikasi yang terbaik yaitu nilai *accuracy* sebesar 80%, *precision* sebesar 53%, *recall* sebesar 47% dan nilai AUC sebesar 0,68. Berdasarkan pada Tabel 4.16 tersebut, *subset* ke-10 merupakan *subset* yang terbaik dengan kombinasi data *training* dan data *testing* yang menghasilkan *confusion matrix* sebagai berikut.

Tabel 4.21 *Confusion matrix* SVM RBF Data Awal

Kelas Aktual	Kelas Prediksi	
	Negatif	Positif
Negatif	115	16
Positif	16	20

Berdasarkan tabel *confusion matrix* diatas dapat diketahui bahwa ketepatan klasifikasi untuk data negatif yang diklasifikasikan benar negatif adalah 115, sedangkan untuk data negatif diklasifikasikan positif adalah 16. Data positif yang diklasifikasikan negatif sebesar 16, dan untuk data *tweet* negatif diklasifikasikan benar negatif adalah 20. Sehingga dapat putuskan bahwa nilai parameter *C* optimum pada kasus klasifikasi pendapat masyarakat terhadap Pemkot Surabaya dengan metode SVM *kernel* RBF adalah 10^2 dengan γ pada nilai 10^{-1} yang menghasilkan nilai AUC sebesar 0,72. Dari substitusi nilai γ pada fungsi *kernel* RBF maka diperoleh fungsi sebagai berikut.

$$K\left(\mathbf{x_1}, \mathbf{x_2}\right) = \exp\left(-0{,}1\left\|\mathbf{x_1}, \mathbf{x_2}\right\|^2\right) \tag{4.1}$$

Berdasarkan fungsi *kernel* RBF tersebut maka dapat didistribusikan pada persamaan *hyperplane* sebagai berikut.

$$f\left(x\right) = \sum_{i=1}^{1169} \alpha_i y_i K\left(\mathbf{x_1}, \mathbf{x_2}\right) - 3{,}09 \tag{4.2}$$

Persamaan yang diperoleh sebagaimana pada Persamaan 4.2 dapat digunakan untuk mengklasifikasikan data selanjutnya, dimana a_i adalah nilai koefisien dari *support vector* dimana nilainya dijelaskan pada Lampiran 11. y_i merupakan label kelas yang memiliki dua nilai yaitu (+1) dan (1). Serta **x** merupakan nilai input yang akan diklasifikasikan.

4.5.1 Metode *Suppor Vector Machine* (SVM) *Kernel* RBF pada Data SMOTE

Selanjutnya dilakukan analisis klasifikasi menggunakan metode Suppport *Vector* Machine (SVM) dengan *kernel* RBF

pada data hasil *oversampling* SMOTE. Telah dilakukan analisis klasifikasi SVM *kernel* RBF pada data hasil *oversampling* sebagaimana pada Lampiran 10, berikut merupakan hasil yang diperoleh dari setiap *subset* 10 *fold* pada masing-masing nilai parameter γ dan parameter C=10. Dipilih parameter C=10 yang ditampilkan, sebab pada parameter ini terdapat nilai optimum dari performa klasifikasinya.

Tabel 4.22 Pemilihan *subset* tebaik SVM RBF data SMOTE

Nilai Parameter (C)	***Subset 10-Folds***	***Accuracy***	***Precision***	***Recall***	**AUC**
10^{-2}	*Subset* ke-1	0.75	0.44	0.62	0.70
	Subset ke-2	0.73	0.41	0.57	0.67
	Subset ke-3	0.82	0.57	0.70	0.78
	Subset ke-4	0.78	0.42	0.73	0.72
	Subset ke-5	0.68	0.30	0.35	0.56
	Subset ke-6	0.55	0.28	0.68	0.60
	Subset ke-7	0.81	0.56	0.65	0.75
	Subset ke-8	0.76	0.46	0.62	0.71
	Subset ke-9	0.82	0.60	0.57	0.73
	Subset ke-10	0.76	0.47	0.78	0.77
Rata-rata		0.75	0.45	0.63	0.70
10^{-1}	*Subset* ke-1	0.78	0.48	0.27	0.59
	Subset ke-2	0.81	0.61	0.38	0.66
	Subset ke-3	0.83	0.72	0.35	0.66
	Subset ke-4	0.80	0.58	0.41	0.66
	Subset ke-5	0.79	0.58	0.19	0.58
	Subset ke-6	0.67	0.31	0.41	0.58
	Subset ke-7	0.82	0.65	0.35	0.65
	Subset ke-8	0.81	0.62	0.35	0.65
	Subset ke-9	0.83	0.70	0.38	0.67
	Subset ke-10	0.83	0.62	0.58	0.74
Rata-rata		0.80	0.59	0.37	0.64
10^{0}	*Subset* ke-1	0.80	0.75	0.16	0.57
	Subset ke-2	0.81	0.86	0.16	0.58
	Subset ke-3	0.83	0.90	0.24	0.62
	Subset ke-4	0.82	0.88	0.19	0.59
	Subset ke-5	0.80	1.00	1.00	0.54
	Subset ke-6	0.74	0.37	0.27	0.57

Nilai Parameter (C)	*Subset 10-Folds*	*Accuracy*	*Precision*	*Recall*	AUC
10^0	*Subset* ke-7	0.81	0.86	0.16	0.58
	Subset ke-8	0.82	0.88	0.19	0.59
	Subset ke-9	0.85	0.93	0.35	0.67
	Subset ke-10	0.84	0.75	0.42	0.69
Rata-rata		0.81	0.82	0.31	0.60
10^1	*Subset* ke-1	0.78	0.00	0.00	0.50
	Subset ke-2	0.78	0.00	0.00	0.50
	Subset ke-3	0.83	0.90	0.24	0.62
	Subset ke-4	0.82	0.88	0.19	0.59
	Subset ke-5	0.80	1.00	1.00	0.54
	Subset ke-6	0.74	0.37	0.27	0.57
	Subset ke-7	0.81	0.86	0.16	0.58
	Subset ke-8	0.82	0.88	0.19	0.59
	Subset ke-9	0.85	0.93	0.35	0.67
	Subset ke-10	0.84	0.75	0.42	0.69
	Subset ke-10	0.81	1.00	0.11	0.56
Rata-rata		0.78	0.20	0.02	0.51
10^2	*Subset* ke-1	0.78	0.00	0.00	0.50
	Subset ke-2	0.78	0.00	0.00	0.50
	Subset ke-3	0.78	0.00	0.00	0.50
	Subset ke-4	0.78	0.00	0.00	0.50
	Subset ke-5	0.78	0.00	0.00	0.50
	Subset ke-6	0.72	0.00	0.00	0.48
	Subset ke-7	0.78	0.00	0.00	0.50
	Subset ke-8	0.78	0.00	0.00	0.50
	Subset ke-9	0.80	1.00	0.11	0.55
	Subset ke-10	0.81	1.00	0.11	0.56
Rata-rata		0.78	0.20	0.02	0.51

nilai *accuracy*, *precision*, *recall* dan AUC dengan metode SVM *kernel* RBF pada data hasil *oversampling* SMOTE, dimana pada nilai parameter 10^2 di *subset* ke-10 memiliki nilai performa klasifikasi yang terbaik yaitu nilai *accuracy* sebesar 46%, *precision* sebesar 47%, *recall* sebesar 89% dan nilai AUC sebesar 0,81. Berdasarkan pada Tabel 4.22 tersebut, *subset* ke-10 merupakan *subset* yang terbaik dengan kombinasi data *training*

dan data *testing* yang menghasilkan *confusion matrix* sebagai berikut.

Tabel 4.23 *Confusion matrix* SVM RBF Data SMOTE

Kelas Aktual	**Kelas Prediksi**	
	Negatif	Positif
Negatif	113	18
Positif	10	26

Berdasarkan tabel *confusion matrix* diatas dapat diketahui bahwa ketepatan klasifikasi untuk data negatif yang diklasifikasikan benar negatif adalah 113, sedangkan untuk data negatif diklasifikasikan positif adalah 18. Data positif yang diklasifikasikan negatif sebesar 10, dan untuk data *tweet* negatif diklasifikasikan benar negatif adalah 26. Sehingga dapat putuskan bahwa nilai parameter *C* dan γ optimum pada kasus klasifikasi pendapat masyarakat terhadap Pemkot Surabaya dengan metode SVM *kernel* RBF adalah C=10 dan $\gamma = 10^{-2}$ dengan perolehan nilai AUC sebesar 0,79. Dari substitusi nilai γ pada fungsi *kernel* RBF data oversampling SMOTE maka diperoleh fungsi sebagai berikut.

$$K\left(\mathbf{x_1}, \mathbf{x_2}\right) = \exp\left(-0,01\left\|\mathbf{x_1}, \mathbf{x_2}\right\|^2\right) \tag{4.3}$$

Berdasarkan fungsi *kernel* RBF tersebut maka dapat didistribusikan pada persamaan *hyperplane* sebagai berikut.

$$f\left(x\right) = \sum_{i=1}^{1169} \alpha_i y_i K\left(\mathbf{x_1}, \mathbf{x_2}\right) - 2,93 \tag{4.4}$$

Persamaan yang diperoleh sebagaimana pada Persamaan 4.4 dapat digunakan untuk mengklasifikasikan data selanjutnya, dimana a_i adalah nilai koefisien dari *support vector* dimana nilainya dijelaskan pada Lampiran 11. y_i merupakan label kelas yang memiliki dua nilai yaitu (+1) dan (1). Serta **x** merupakan nilai input yang akan diklasifikasikan. Selanjutnya akan dilakukan perbandingan antara performa klasifikasi data awal

dengan data yang telah dilakukan *oversampling* SMOTE pada metode SVM *kernel Linear*. Perbandingan performa klasifikasi dilakukan dengan cara membandingkan rata-rata performa klasifikasi setiap *subset fold* pada nilai parameter terbaik yaitu 10^2, sehingga diperoleh ringakasan sebagai berikut.

Tabel 4.24 Perbandingan Performa Klasifikasi SVM RBF

Data	*Accuracy*	*Precision*	*Recall*	AUC
Data Awal	0.79	0.52	0.33	0.64
Data SMOTE	0.75	0.45	0.63	0.70

Berdasarkan pada Tabel 4.24 dapat diketahui bahwa data hasil *oversampling* SMOTE menghasilkan rata-rata performa klasifikasi lebih baik dari pada data awal.

4.6 Metode Klasifikasi Regresi Logistik

Metode klasifikasi Regresi Logistik merupakan metode statistika yang klasik yang akan dibandingkan dengan metode klasifikasi NBC dan SVM. *Output* yang akan dihasilkan dari metode klasifikasi Regresi Logistik adalah model regresi. Sebagaimana pada metode NBC dan SVM yang telah dilakukan, data yang digunakan adalah data pendapat masyarakat terhadap Pemkot Surabaya berdasarkan Media Sosial Twitter. Dari data pendapat tersebut selanjutnya dilakukan pengklasifikasin sehingga diketahui sentimen yang dikandung dalam setiap *tweet*nya. Dari seluruh data yang digunakan, data yang mengandung sentimen negatif lebih dominan dengan prosentase 78,1% daripada *tweet* yang mengandung sentimen positif. Sehingga dilakukan *balancing* data menggunakan *oversampling* SMOTE. Berdasarkan data awal dan data hasil SMOTE tersebut dilakukan perhitungan sehingga diperoleh model dan performa klasifikasi dengan menggunakan metode Regresi Logistik sebagai berikut.

4.6.1 Metode Regresi Logistik Data Awal

Klasifikasi menggunakan metode Regresi Logistik menghasilkan sebuah model, dimana model tersebut dapat digunakan untuk mengklasifikasikan data-data selanjutnya.

Model tersebut akan diperoleh dari proporsi data *training* dan *testing* yang terbaik. Cara menentukan pembagian data tersebut adalah yang terbaik yaitu dengan menghitung performa klasifikasinya. Pembagia data *training* dan data *testing* dilakukan menggunakan *cross validation* 10-*fold*s, sehingga dapat diperoleh hasil setiap *subset* 10 *fold* adalah sebagai beirkut.

Tabel 4.25 Pemilihan *Subset* Tebaik Regresi Logistik data Awal

Subset 10-Folds	***Accuracy***	***Precision***	***Recall***	**AUC**
Subset ke-1	0.78	0.00	0.00	0.50
Subset ke-2	0.79	1.00	0.03	0.51
Subset ke-3	0.82	0.56	0.76	0.80
Subset ke-4	0.73	0.43	0.73	0.73
Subset ke-5	0.70	0.33	0.35	0.58
Subset ke-6	0.79	1.00	0.05	0.58
Subset ke-7	0.79	1.00	0.05	0.73
Subset ke-8	0.66	1.00	0.03	0.70
Subset ke-9	0.80	1.00	0.08	0.54
Subset ke-10	0.80	0.75	0.08	0.54
Rata-rata	0.77	0.71	0.22	0.62

Sebagaimana pada Tabel 4.25 dapat pada hasil nilai *accuracy*, *precision*, *recall* dan AUC dengan metode regresi logistik pada data awal. Dimana pada *subset* ke-3 memiliki nilai performa klasifikasi yang tertinggi yaitu nilai *accuracy* sebesar 82%, *precision* sebesar 56%, *recall* sebesar 76% dan nilai AUC sebesar 0,80. Nilai *precision* dan *recall* disini sangat kecil karena dapat diketahui bahwa *tweet* yang mengandung sentimen positif adalah minoritas, dimana perhitungan *precision* dan *recall* menghitung ketepatan untuk data positif. Berdasarkan pada Tabel 4.25 tersebut, dipilih salah satu dari ketiga *subset* yang menghasilkan nilai performa klasifikasi yang sama yaitu *subset* ke-3. Dari pembagian data *training* dan *testing* pada *subset* ke-3 selanjutnya dibuat model regresi sebagai berikut.

$$\hat{\pi}(x) = \frac{\exp\left(-8.69\times10^{34} - 2.05\times10^{34} X_1 + \ldots + 8.58\times10^{39} X_{3689}\right)}{1+\exp\left(-8.69\times10^{34} - 2.05\times10^{34} X_1 + \ldots + 8.58\times10^{39} X_{3689}\right)} \quad (4.5)$$

Berdasarkan model tersebut kemudian dapat dihasilkan *confusion matrix*, sehingga dapat diketahui performa ketepatan klasifikasinya.

Tabel 4.26 *Confusion matrix* Regresi Logistik Data Awal

Kelas Aktual	**Kelas Prediksi**	
	Negatif	Positif
Negatif	132	0
Positif	34	3

Berdasarkan Tabel *confusion matrix* diatas dapat diketahui bahwa ketepatan klasifikasi untuk data negatif yang diklasifikasikan benar negatif adalah 132, sedangkan tidak ada data negatif diklasifikasikan positif. Data positif yang diklasifikasikan negatif sebesar 34, dan untuk data *tweet* negatif diklasifikasikan benar negatif adalah 3.

4.6.2 Metode Regresi Logistik Data SMOTE

Metode *oversampling* SMOTE merupakan metode *balancing* data dengan cara menyeimbangkan jumlah data pada minoritas dengan data mayoritas. Hal ini dilakukan dengan cara replika secara acak sehingga diperoleh jumlah yang seimbang. Telah dilakukan analisis regresi logistik dengan data hasil *oversampling* SMOTE menggunakan *cross validation* 10 *fold* sehingga diperoleh performa klasifikasi untuk masing-masing *subset fold*nya adalah sebagai berikut.

Tabel 4.27 Pemilihan *Subset* Tebaik Regresi Logistik Data SMOTE

Subset* 10-*Folds	***Accuracy***	***Precision***	***Recall***	**AUC**
Subset ke-1	0.74	0.43	0.54	0.67
Subset ke-2	0.75	0.44	0.62	0.70
Subset ke-3	0.82	0.56	0.76	0.80
Subset ke-4	0.73	0.43	0.73	0.73
Subset ke-5	0.70	0.33	0.35	0.58
Subset ke-6	0.54	0.27	0.65	0.58
Subset ke-7	0.79	0.51	0.62	0.73
Subset ke-8	0.76	0.46	0.59	0.70
Subset ke-9	0.80	0.53	0.62	0.73
Subset ke-10	0.78	0.49	0.89	0.82
Rata-rata	0.74	0.45	0.64	0.70

Sebagaimana pada Tabel 4.27 dapat diketahui hasil nilai *accuracy*, *precision*, *recall* dan AUC dengan metode NBC pada data hasil *oversampling* SMOTE. Dimana pada *subset* ke-10 memiliki nilai performa klasifikasi yang terbaik yaitu nilai *accuracy* sebesar 78%, *precision* sebesar 49%, *recall* sebesar 89% dan nilai AUC sebesar 0,82. Berdasarkan pada Tabel 4.23 tersebut, *subset* ke-10 merupakan *subset* yang terbaik dan dengan kombinasi data *training* dan data *testing* pada *subset* ini akan dilakukan permodelan sehingga pada data selanjutnya dapat langsung menggunakan model tersebut. Model regresi berdasarkan pada *subset* ke-10 data hasil *oversampling* SMOTE adalah sebagai berikut.

$$\hat{\pi}(x)=\frac{\exp\left(1.6\times10^{25}-1.39\times10^{29}X_1+\ldots+3.64\times10^{18}X_{3689}\right)}{1+\exp\left(1.6\times10^{25}-1.39\times10^{29}X_1+\ldots+3.64\times10^{18}X_{3689}\right)} \quad (4.6)$$

Berdasarkan model tersebut kemudian dapat dihasilkan *confusion matrix*, sehingga dapat diketahui performa ketepatan klasifikasinya

Tabel 4.28 *Confusion matrix* Regresi Logistik Data SMOTE

Kelas Aktual	**Kelas Prediksi**	
	Negatif	Positif
Negatif	130	1
Positif	33	3

Berdasarkan Tabel *confusion matrix* diatas dapat diketahui bahwa ketepatan klasifikasi untuk data negatif yang diklasifikasikan benar negatif adalah 130, sedangkan untuk data negatif diklasifikasikan positif adalah 1. Data positif yang diklasifikasikan negatif sebesar 33, dan untuk data *tweet* negatif diklasifikasikan benar negatif adalah 3.

Selanjutnya akan dilakukan perbandingan antara performa klasifikasi data awal dengan data yang telah dilakukan *oversampling* SMOTE pada metode regresi logistik. Perbandingan performa klasifikasi dilakukan dengan

membandingkan rata-rata performa klasifikasi setiap *subset fold*, sehingga diperoleh ringakasan sebagai berikut.

Tabel 4.29 Perbandingan Performa Klasifikasi Regresi Logisitik

Data	*Accuracy*	*Precision*	*Recall*	AUC
Data Awal	0.77	0.71	0.22	0.62
Data SMOTE	0.74	0.45	0.64	0.70

Berdasarkan pada Tabel 4.29 dapat diketahui bahwa data hasil *oversampling* SMOTE menghasilkan rata-rata performa klasifikasi lebih baik dari pada data awal.

4.7 Penentuan Metode Klasifikasi Terbaik

Telah diperoleh nilai ketepatan klasifikasi dari parameter terbaik di setiap metode klasifikasi. Selanjutnya akan dilakukan perbandingan nilai rata-rata performa klasifikasi untuk memperoleh metode klasifikasi yang terbaik. Tabel 4.30 merupakan hasil ringkasan nilai rata-rata *accuracy*, *precision*, *recall*, dan AUC untuk tiap metode dan data yang digunakan, sehingga dapat diketahui metode yang menghasilkan performa klasifikasi paling baik.

Tabel 4.30 Performa Klasifikasi Semua Metode

Metode	Data Awal				Data SMOTE			
	A	***P***	***R***	**AUC**	***A***	***P***	***R***	**AUC**
NBC	0.77	0.43	0.11	0.53	0.74	0.45	0.58	0.68
SVM *Linear*	0.73	0.58	0.63	0.65	0.71	0.4	0.63	0.68
SVM RBF	0.79	0.54	0.38	0.64	0.75	0.45	0.63	0.70
Regresi Logistik	0.77	0.71	0.22	0.62	0.74	0.45	0.64	0.70

Berdasarkan pada ringkasan hasil rata-rata performa klasifikasi dapat diketahui bahwa metode yang menghasilkan performa klasifikasi terbaik adalah metode SVM *kernel* RBF pada kasus klasifikasi pendapat masyarakat terhadap Pemkot Surabaya.

4.8 *Social Network Analysis*

Social Network Analysis (SNA) merupakan analisis yang digunakan untuk mengetahui pemilik akun twitter yang memberi pengaruh besar pada topik ini yaitu Pemkot Surabaya. Sangat

banyak akun twitter yang turut berkontribusi dalam pemberian komentar mengenai Pemkot Surabaya, pada analisis ini akan dilihat jaringan-jaringan dari semua akun dan interaksi antar akun tersebut. Sehingga dapat diketahui akun yang paling sering berinteraksi dan berkontribusi. Pada analisis ini data *tweet* yang digunakan merupakan data keseluruhan dari data hasil *crawling* tanpa mempertimbangkan sentimen positif atau negatifnya, sebab hal yang akan menjadi fokus utama adalah jaringan antar pemilik akun twitter.

Terdapat dua variabel penting pada analisis SNA yaitu *node* dan egde. *Node* adalah titik akun pengguna twitter, sedangkan egde merupakan interaksi antar *node* tersebut. Jumlah *node* dan *edge* ini sangat berhubungan dengan angka *degree range*, dimana *degree range* ini adalah angka interaksi minimal yang dilakukan oleh antar *node*. Untuk lebih jelasnya, Tabel 4.31 akan menampilkan hubungan antara jumlah *degree range*, *node*, dan *edge*.

Tabel 4.31 Hubungan *degree range*, *node*, dan *edge*

Degree Range	***Nodes***	***Edges***
0	26.974	27.169
3	3917	16.598
5	2200	11.854
10	910	6166

Berdasarkan pada Tabel 4.31 dapat diketahui bahwa jika menggunakan *degree range* awal yaitu 0 maka jumlah *node* adalah 26.974 dan jumlah *edge* adalah 27.169. Jumlah *node* dan *edge* tersebut sangatlah besar dan apabila dilanjutkan pada membuatan akan menghasilkan gambar yang tidak interpretatif. Oleh sebab itu dilakukan filter *degree range* untuk mempermudah dalam melihat hubungan antar pemilik akun twitter secara visual. Dilakukan filter *degree range* hingga diperoleh *degree range* yang dirasa optimum yaitu 10 dengan 910 *nodes* dan 6166 *edges*. 910 *nodes* ini berarti bahwa jumlah pemilik akun twitter yang akan digunakan dalam analisis adalah 910 akun. Sedangkan 6166 *edges* ini berarti bahwa interaksi yang akan dilakukan analisis

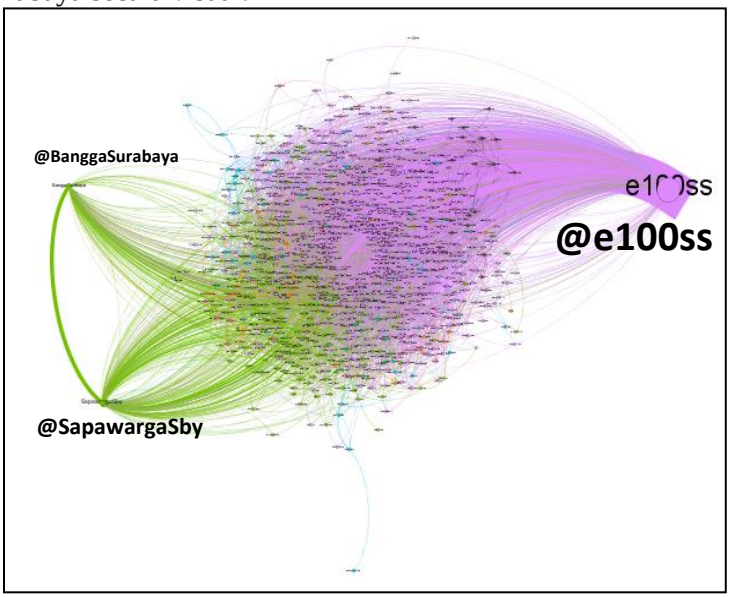
sebanyak 6166 interaksi. Jumlah *degree range*, *node*, dan *edge* yang telah ditentukan tersebut selanjutnya dilakukan analisis menggunakan tiga jenis *centrality* yaitu *degree centrality*, *closeness centrality* dan *betweenness centrality*. Hasil perhitungan dari analisis SNA dengan ketiga metode *centrality* tersebut telah ditabelkan sebagaimana pada Tabel 4.32. Dimana pada tabel tersebut dapat diketahui nama akun-akun yang sangat berpengaruh pada pendapat masyarakat kota Surabaya berdasarkan media sosial twitter.

Tabel 4.32 Analisis *Centrality*

Centrality	***Username***	***Score***
Degree Centrality	@e100ss	761
	@SapawargaSby	226
	@BjunaediIrianto	119
Closeness Centrality	@Bbksdajatim	1,00
	@Sein_dra	1,00
	@SapawargaSby	0,44
Betweenness Centrality	@e100ss	390234,8
	@SapawargaSby	159929,5
	@BjunaediIrianto	93815,3

Berdasarkan pada hasil analisis SNA di Tabel 4.32 dapat diketahui bahwa tiga akun yang sangat berpengaruh berdasarkan pada metode analisis yaitu *degree centrality*, *closeness centrality*, *betweenness centrality*. Dari ketiga metode analisis tersebut memberikan hasil yang berbeda. Pada *degree centrality* dan *betweenness centrality* memberikan nama akun pada urutan pertama yang sama yaitu akun @e100ss dimana akun ini adalah akun resmi dari Radio Suara Surabaya. Sehingga menurut perhitungan dapat diketahui bahwa akun @e100ss sangat berpengaruh terhadap topik Pemkot Surabaya. Menurut metode analisis *degree centrality* dan *betweenness centrality*, akun selanjutnya yang berpengaruh pada topik ini adalah akun @SapawargaSby dimana diketahui akun tersebut adalah akun resmi dari Pemkot Surabaya. Dan akun yang yang berpengaruh pada urutan ketiga adalah @BjunaediIrianto yang mana akun

tersebut adalah akun masyarakat kota Surabaya yang sering melakukan interaksi mengenai topik Pemkot Surabaya. Hasil yang berbeda diberikan oleh metode analisis SNA *closeness centrality* yaitu pada urutan pertama akun @Bbksdajatim, urutan kedua adalah akun @Sein_dra dan pada urutan ketiga adalah @SapawargaSby. Hasil dari analisis SNA ini juga dapat dilakukan berdasarkan *graph* sehingga dapat dilihat secara jelas akun manakah yang sangat berpengruh terhadap topik Pemkot Surabaya secara visual.



Gambar 4.10 *Social Network Analysis*

Berdasarkan visualisasi SNA pada Gambar 4.10 dapat diketahui jika akun yang memiliki warna yang dominan maka akun tersebut memiliki kontribusi yang lebih pada topik ini. Dapat diketahui bahwa akun yang berwarna ungu dan jaring yang berwarna ungu sangat dominan, hal ini berarti @e100ss merupakan akun yang sangat berpengaruh daripada akun-akun

yang lainnya. Warna dominan yang kedua adalah warna hijau disisi kiri, dimana terdapat dua akun yang menjadi tujuan ataupun sumber dari jaring yang berwarna hijau tersebut. Kedua akun tersebut adalah @SapawargaSby yang merupakan akun resmi dari Pemkot Surabaya dan @BanggaSurabaya yang merupakan akun resmi dari Humas Kota Surabaya. Berdasarkan jaring yang berwarna hijau dapat diketahui bahwa akun @SapawargaSby memiliki kontribusi yang kebih daripada @BanggaSurabaya, serta kedua akun tersebut memiliki jaringan yang cukup besar pula. Berdasarkan penjelasan yang telah dipaparkan maka dapat disimpulkan bahwa urutan tiga akun twitter yang berpengaruh terhadap topik Pemkot Surabaya adalah @e100ss, @SapawargaSby, dan yang terakhir @BanggaSurabaya secara visualisasi.

BAB V
PENUTUP

5.1 Kesimpulan

Berdasarkan analisis dan pembahasan yang telah dijelaskan sebelumnya, maka kesimpulan yang dapat diambil adalah sebagai berikut.

1. Masyarakat lebih aktif mengemukakan pendapatnya kepada akun Radio Suara Surabaya (@e100ss) daripada akun resmi Pemerintah Kota Surabaya (@SapawargaSby), serta pendapat yang dikemukakan didominasi oleh keluhan atau sentimen negatif yaitu sebesar 78,1% dan untuk pendapat yang mengandung sentimen positif sebesar 21,9%.
2. Jumlah *tweet* yang digunakan adalah 1687 *tweet* sedangkan jumlah kata kunci dari hasil praroses data adalah 3689 kata.
3. Masyarakat sangat mengapresiasi kepada pihak polisi lalu lintas dalam pengaturan jalan, akan tetapi masih banyak dari masyarakat yang mengeluhkan macet di Surabaya.
4. Hasil ketepatan klasifikasi SVM *kernel RBF* lebih baik dari pada Metode klasifikasi yang lainnya yaitu *Naïve Bayes Classifier*, dan Regresi Logistik.
5. Urutan tiga akun twitter yang berpengaruh terhadap topik Pemerintah Kota Surabaya adalah @e100ss, @SapawargaSby, dan yang terakhir @BanggaSurabaya.

5.2 Saran

Saran yang dapat diberikan dari hasil penelitian ini adalah sebagai berikut.

1. Menjadikan hasil penelitian ini informasi tambahan supaya Pemerintah Kota Surabaya menjadikan masalah kemacetan di Surabaya sebagai masalah penting yang harus dicarikan solusinya, karena mayoritas masyarakat

mengeluhkan tentang kemacetan melalui sosial media twitter.

2. Untuk penelitian selanjutnya, dapat melakukan normalisasi pada variabel prediktor / kata, sehingga dapat menggantikan singkatan menjadi kata yang sebenarnya. Serta melakukan reduksi kata untuk beberapa kata dengan arti yang sama, sehingga frekuensi yang diperoleh akan lebih besar.

DAFTAR PUSTAKA

Agresti, A. (2007). *An Introduction to Categorical Data Analysis.* New York: John Wiley & Sons.

Ardianto, A. F. (2017, Januari 19). *www.google.com*. Retrieved from http://www.beritajatim.com/: http://www.beritajatim.com/politik_pemerintahan/287875/inilah_prestasi_anyar_kota_surabaya.html

Ariadi, D., & Fithriasari, K. (2015). Klasifikasi berita Indonesia Menggunakan Bayesian Classification dan *Support Vector Machine* dengan Confix Stripping Stemmer. *Jurnal Sains dan Seni ITS*, Vol.4, No.2.

Asiyah, S. N., & Fithriasari, K. (2016). Klasifikasi Berita Online Menggunkaan Metode *Support Vector Machine* dan K-Nearest Neighbor. *Jurnal Sains dan Seni ITS*, Vol.5, No.2.

Bekkar, M., Djemaa, D. K., & Alitouche, D. A. (2013). Evaluation Measure for Models Assument over Imbalanced Data Sets. *Journal of Internation Engineering and* , 27-36.

Bing, L. (2010). *Handbook of Natural Language Processing Second Edition.* Boca Raton: CRC Press.

Buntoro, G. A., Adji, T. B., & Purnamasari, A. E. (2014). Sentiment Analysis Twitter dengan Kombinasi Lexicon Based dan Double Propagation., (pp. 39-43). Yogyakarta.

Byun, H., & Lee, S.-W. (2003). A Survey on Pattern Recognition Applications of *Support Vector Machine*ss. *International Journal of Pattern Recognition and Artificial Intelligence*, 459-486.

Castella, Q., & Sutton, C. (2014). *Wrod Storms: Multiples of Word Clouds for Visual Comparison of Documents.* Seoul: University of Edinburgh.

Chawla, N. V. (2005). *Data Mining and Knowledge Discovery Handbook.* USA: Univercity of Notre Dame Press.

Cheliotis, G. (2010). *Social Network Analysis (SNA) Including a Tutorial on Concepts and Methods.* Singapore:

Communications and New Media, National University of Singapore.

Dragut, E., Fang, F., Sistla, P., & Yu, C. (2009). *Stop Word and Related Problems in Web Interface Integration.* Chicago: University of Illinois.

Feldman, R., & James, S. (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*, 138-144.

Gorunescu, F. (2011). *Data Mining Concepts, Models and Techniques.* Berlin: Spinger-Verlag.

Hardle, W. K., Prastyo, D. D., & Hafner, C. (2014). *Support Vector Machines with Evolutionary Feature Selection for Defult Prediction.* Oxford University Press.

Hosmer, D. W., & Lemeshow, S. (2000). *Applied Logistic Regression.* New York: John Wiley & Sons, Inc.

Hotho, A., Nurnburger, A., & Paass, G. (2005). *Brief Survey of Text Mining.* Kassel: University of Kassel.

Mariati, R. (2012). *Pengadaan Barang dan Jasa Pemerintah di Lingkungan Pemerintah Kabupaten Kutai Timur Provinsi Kalimantan Timur.* Yogyakarta: Universitas Atma Jaya Yogyakarta.

Media, K. S. (2018, Februari 26). *http://www.google.com.* Retrieved from http://www.suarasurabaya.net/: http://www.suarasurabaya.net/ssmedia/

Nugroho, A. S., Witarto, A. B., & Handoko, D. (2003). *Support Vector Machine*: Teori dan Aplikasinya dalam Bioinformatika.

Prakasita, R., & Fithriasari, K. (2015). Klasifikasi Kesejahteraan Rumah Tangga di Provinsi Papua dengan Metode Regresi Logistik dan *Support Vector Machine*. *Jurnal Sains dan Seni ITS*, Vol.4, No.2.

Setyani, N. I. (2013). *Pengguna Media Sosial sebagai Sarana Komunikasi bagi Komunitas.* Surakarta: Universitas Sebelas Maret.

Siang, J. J. (2005). *Jaringan Syaraf Tiruan dan Pemograman menggunakan Matlab.* Yogyakarta: Andi Publisher.

Tim APJII. (2016). Saatnya Jadi Pokok Perhatian Pemerintah dan Indistri. *Buletin APJII*, 1.

Weiss, S. M., Indurkhya, N., Zhang, T., & Damerau, F. J. (2005). *Text Mining Predictive Methods for Analyzing Unstructures Information.* New York: Spinger Science+Business Media. Inc.

(Halaman ini sengaja dikosongkan)

LAMPIRAN

Lampiran 1. *Crawling* Data *soft ware* R

```
library(twitteR)
options(httr_oauth_cache=T)
rconsumer_key <- 'JgtAKEJUEsGq2B6h36G7bpRYm'
consumer_secret <-
'v2ZDIy7LuIKW9R1fwhFZFS4TyYbYCURVQWmxMVAwK7Fo4KNHQJ'
access_token <- '72809642-BmENMhkPCNpZrHv7c9mCSphdHDhJz7Ixb3kbVMDKA'
access_secret <- '2OI6M8UUeg7uavSqa4BO08G9Nrtn2Sb1RkG7SewMESXd5'
setup_twitter_oauth(consumer_key, consumer_secret, access_token,
access_secret)
ss <- searchTwitter('e100ss', lang="id", n=15000,
resultType="recent")
write.csv(twListToDF(ss), file="craw20ss.csv")
pemkot <- searchTwitter('SapawargaSby', lang="id", n=15000,
resultType="recent")
write.csv(twListToDF(pemkot), file="craw20sapa.csv")
```

Lampiran 2. Data *Tweet*

DATA HASIL CRAWLING @SapawargaSby								
No	Text	…	replyTo SN	created	…	screen Name	…	latitude
1	Ayo mampir ke GAT …	…	antobudisaputra	28/08/2017 00:06	…	rizqoo	…	NA
2	RT @PMI_Surabaya …	…	NA	29/08/2017 00:59	…	sapawargasby	…	NA
⋮	⋮		⋮	⋮		⋮		⋮
3437	Ntap. @naomidl_https://…	…	NA	03/03/2018 02:25	…	o2o598	…	NA
⋮	⋮		⋮	⋮		⋮		⋮
7129	@ceritasby @TICSby…	…	ceritaSby	07/05/2018 05:00	…	ceritasby	…	NA
7130	RT @nanangsigit1: …	…	NA	07/05/2018 05:12	…	nanangsigit	…	NA

DATA HASIL CRAWLING @e100ss								
No	Text	…	replyTo SN	created	…	screen Name	…	latitude
1	@e100ss Mohon dinas …	…	e100ss	28/08/2017 23:45	…	jonathan taffy	…	NA
2	@e100ss merambat parah …	…	e100ss	29/08/2017 23:46	…	motorrio blog	…	NA
⋮	⋮		⋮	⋮		⋮		⋮
61957	Ntap. @naomidl_htt	…	NA	03/03/2018 02:25	…	o2o598	…	NA

	ps://…							
⋮	⋮		⋮	⋮		⋮		⋮
116061	@e100ss @sits_dishubsby …	…	e100ss	07/05/2018 04:05	…	oktarir	…	NA
116062	RT@15DEATH: @e100ss…	…	NA	07/05/2018 04:05	…	afinkusani	…	NA

DATA HASIL CRAWLING GABUNGAN								
No	text	…	replyTo SN	created	…	screen Name	…	latitude
1	@e100ss Mohon dinas …	…	e100ss	28/08/2017 23:45	…	jonathan taffy	…	NA
2	@e100ss merambat parah …	…	e100ss	29/08/2017 23:46	…	motorrio blog	…	NA
⋮	⋮		⋮	⋮		⋮		⋮
61596	Minta tolong utk diberikan…	…	NA	26/01/2018 11:12	…	sylaahs	…	NA
⋮	⋮		⋮	⋮		⋮		⋮
123191	@e100ss @sits_dishubsby …	…	e100ss	07/05/2018 04:05	…	oktarir	…	NA
123192	RT@15DEATH: @e100ss…	…	NA	07/05/2018 04:05	…	afinkusani	…	NA

DATA SENTIMEN GABUNGAN		
No	sentimen	text
1	0	@e100ss ada proses evakuasi bis mira terperosok ke kali, menyebabkan macet panjang
2	0	@e100ss deretan PKL disebelah timur pos jemur makan bahu jalan Sangat Menggaggu saying dibiarkan saja sama… https://t.co/oMexCS22qN
⋮	⋮	⋮
985	0	@e100ss jd agak kuatir tar pas mau urus perpanjangan paspor. <ed><U+00A0><U+00BD><ed><U+00B8><U+0085>. Bbrp kali dgr info sering begini. #imigrasi https://t.co/LvQjHBkySc
⋮	⋮	⋮
1686	0	@SapawargaSby Alhamdulillah,dengan adanya akun @SapawargaSby sangat membantu dan respon cepat,terima kasih :)
1687	0	@SapawargaSby Slmt malam, bgaimana pelaporan sy ttg kinerja kelurahan bulak? Apakah sdh d tindak lanjuti?

Lampiran 3. *Input* Python

```
import pandas as pd
import string
import nltk
from nltk.tokenize import word_tokenize
import re
import sys
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
data= pd.read_csv('dataTA.csv', engine='python')
pre_tweet=data['text']
print (data)
```

```
     sentimen                          text
0            0  @e100ss ada proses evakuasi bis mira yg...
1            0  @e100ss deretan PKL disebelah timur...
2            0  @e100ss @SapawargaSby @SatgasPungli...
                                  ⋮
1684         0  @SapawargaSby @BanggaSurabaya Percuma...
1685         1  @SapawargaSby Alhamdulillah,dengan...
1686         0  @SapawargaSby Slmt malam, bgaimana...
```

Lampiran 4. Praproses Data

Clear Link

```
dataclearlink=[]
for line in pre_tweet:
    result=re.sub(r"http\S+","",line)
    dataclearlink.append(result)
```

```
[' ada proses evakuasi bis mira yg terperosok ke kali, menyebabkan
macet panjang ', … '@SapawargaSby Slmt malam, bgaimana pelaporan
sy ttg kinerja kelurahan bulak? Apakah sdh d tindak lanjuti?']
```

Clear #

```
dataclearhastag=[]
for line in dataclearlink:
    result=re.sub(r"#\S+","",line)
    dataclearhastag.append(result)
```

```
[' ada proses evakuasi bis mira yg terperosok ke kali, menyebabkan
macet panjang ', … '@SapawargaSby Slmt malam, bgaimana pelaporan
sy ttg kinerja kelurahan bulak? Apakah sdh d tindak lanjuti?']
```

Casefolding

```
datalower=[]
for line in dataclearhastag:
    a=line.lower()
    datalower.append(a)
```

```
[' ada proses evakuasi bis mira yg terperosok ke kali, menyebabkan
macet panjang ', … ' slmt malam, bgaimana pelaporan sy ttg kinerja
kelurahan bulak? apakah sdh d tindak lanjuti?']
```

Stemming

```
factory=StemmerFactory()
stemmer=factory.create_stemmer()
datastemmed=map(lambda x: stemmer.stem(x), datalower)
databersih=map(lambda x: x.translate(str.maketrans('', '',
```

```
string.punctuation)), datastemmed)
databersih=list(databersih)
```

```
['ada proses evakuasi bis mira yg perosok ke kali sebab macet
panjang', … 'slmt malam bgaimana lapor sy ttg kerja lurah bulak
apakah sdh d tindak lanjut']
```

Stopwords

```
stopwords=open('stopwords.txt', 'r').read()

satudata=[]
datafinal=[]
df=[]
for line in databersih:
  wo = word_tokenize(line)
  wo = [word for word in wo if not word in stopwords and not
word[0].isdigit()]
  datafinal.append(wo)
  df.append(" ".join(wo))
for l in datafinal:
  satudata+= l
final={v: satudata.count(v) for v in set(satudata)}
import csv
```

```
['proses evakuasi mira perosok macet', … 'slmt malam bgaimana
lapor ttg kerja lurah bulak tindak lanjut']
```

Count*Vectorizer*

```
from pandas import DataFrame
from sklearn.feature_extraction.text import CountVectorizer

Y = data['sentimen']
Y_nya = pd.DataFrame(Y)

vect = CountVectorizer(min_df=0., max_df=1.0)
X = vect.fit_transform(df)
X_=DataFrame(X.A, columns=vect.get_feature_names())
pembobotan = TfidfTransformer(use_idf=True).fit_transform(X)
hasil=DataFrame(pembobotan.A, columns=vect.get_feature_names())
```

```
[[ 0.  0.  0. ...,  0.  0.  0.]
 [ 0.  0.  0. ...,  0.  0.  0.]
 [ 0.  0.  0. ...,  0.  0.  0.]
 ...,
 [ 0.  0.  0. ...,  0.  0.  0.]
 [ 0.  0.  0. ...,  0.  0.  0.]
 [ 0.  0.  0. ...,  0.  0.  0.]]
```

Tfidf

```
from sklearn.feature_extraction.text import TfidfTransformer

tfidf = TfidfTransformer(use_idf=True).fit_transform(X_)
tfidf_nya = (tfidf.toarray())

X_nya = tfidf_nya
tf=DataFrame(tfidf.A, columns=vect.get_feature_names())
print (tf)
```

```
      aamiin  abai  … zigzag  zigzah  zona
0        0.0   0.0  … 0.0       0.0     0.0
1        0.0   0.0  … 0.0       0.0     0.0
2        0.0   0.0  … 0.0       0.0     0.0
                                ⋮
3371     0.0   0.0  … 0.0       0.0     0.0
3372     0.0   0.0  … 0.0       0.0     0.0
3373     0.0   0.0  … 0.0       0.0     0.0
```

Lampiran 5. Karakteristik Data

Plot Perbandingan Jumlah *Tweet* Antar Akun

```
import matplotlib.pyplot as plt

labels = 'E100ss', 'SapawargaSby'
sizes = [114912, 7100]
colors = ['yellowgreen', 'silver']
explode = (0.1, 0)  # explode 1st slice
plt.pie(sizes, explode=explode, labels=labels, colors=colors,
        autopct='%1.1f%%', shadow=True, startangle=140)
plt.axis('equal')
plt.show()
```

Plot Perbandingan Jumlah *Tweet* Mengandung Sentimen

```
labels = ['Sentimen', 'Tidak']
sizes = [1531, 113381]
colors = ['yellowgreen', 'gold']
patches, texts = plt.pie(sizes, colors=colors, startangle=120)
plt.legend(patches, labels, loc="best")
plt.axis('equal')
plt.tight_layout()
plt.show()
```

Plot Perbandingan Jumlah *Tweet* Positif dan Negatif

```
labels = 'Positif', 'Negatif'
sizes = [a, b]
colors = ['lightskyblue','lightcoral', ]
explode = (0, 0)  # explode a slice if required

plt.pie(sizes, explode=explode, labels=labels, colors=colors,
        autopct='%1.1f%%', shadow=True)

centre_circle = plt.Circle((0,0),0.75,color='black',
fc='white',linewidth=1.25)
fig = plt.gcf()
fig.gca().add_artist(centre_circle)

plt.axis('equal')
plt.show()
plt.savefig('.png')
```

Lampiran 6. *Word Cloud* Gabung

```
import pandas as pd
import string
import nltk
from nltk.tokenize import word_tokenize
import re
import sys
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
data= pd.read_csv('dataTA.csv', engine='python')
pre_tweet=data['text']
print (data)
```

```
     sentimen                          text
0            0  @e100ss ada proses evakuasi bis mira yg...
1            0  @e100ss deretan PKL disebelah timur...
2            0  @e100ss @SapawargaSby @SatgasPungli...
                                    ⋮
1684         0  @SapawargaSby @BanggaSurabaya Percuma...
1685         1  @SapawargaSby Alhamdulillah,dengan...
1686         0  @SapawargaSby Slmt malam, bgaimana...
```

Split Data

```
data_pos = data [ data['sentimen'] == 1 ]
data_pos = data_pos ['text']

data_neg = data [ data['sentimen'] == 0 ]
data_neg = data_neg ['text']
```

```
3       @e100ss  Pasti untuk kebaikan warga kota sby, ...
4       @e100ss APA &amp; BAGAIMANA PENDAPAT KAWAN2? T...
5       @e100ss  ada bus mira masuk rumah warga di ara...
                                    ⋮
1680    @SapawargaSby Oke min...suwun infone....<ed><U...
1681    @SapawargaSby huaaaa akhirnya dijelaskan detai...
1685    @SapawargaSby Alhamdulillah,dengan adanya akun...
```

```
0       @e100ss ada proses evakuasi bis mira yg...
1       @e100ss deretan PKL disebelah timur kantor pos...
2       @e100ss @SapawargaSby @SatgasPungli @SaberPung...
                                    ⋮
1683    @SapawargaSby banyak pedagang liar yg berjuala...
1684    @SapawargaSby @BanggaSurabaya Percuma bro gawe...
1686    @SapawargaSby Slmt malam, bgaimana pelaporan s...
```

Clear Link

```
dataclear_pos=[]
for line in data_pos:
    result=re.sub(r"http\S+","",line)
    dataclear_pos.append(result)

dataclear_neg=[]
for line in data_neg:
    result=re.sub(r"http\S+","",line)
    dataclear_neg.append(result)
```

```
['@e100ss  Pasti untuk kebaikan warga kota sby, pemikiran bagi yg
memahami prosedur. dan pasti ada pemikiran yg lain nya', …
'@SapawargaSby Alhamdulillah,dengan adanya akun @SapawargaSby
```

```
sangat membantu dan respon cepat,terima kasih :)']
```

```
[' ada proses evakuasi bis mira yg terperosok ke kali, menyebabkan
macet panjang ', … '@SapawargaSby Slmt malam, bgaimana pelaporan
sy ttg kinerja kelurahan bulak? Apakah sdh d tindak lanjuti?']
```

Clear #

```
Dataclearhastag_pos=[]
for line in dataclear_pos:
    result=re.sub(r"#\S+","",line)
    dataclearhastag_pos.append(result)

Dataclearhastag_neg=[]
for line in dataclear_neg:
    result=re.sub(r"#\S+","",line)
    dataclearhastag_neg.append(result)
```

```
['@el00ss  Pasti untuk kebaikan warga kota sby, pemikiran bagi yg
memahami prosedur. dan pasti ada pemikiran yg lain nya', …
'@SapawargaSby Alhamdulillah,dengan adanya akun @SapawargaSby
sangat membantu dan respon cepat,terima kasih :)']
```

```
[' ada proses evakuasi bis mira yg terperosok ke kali, menyebabkan
macet panjang ', … '@SapawargaSby Slmt malam, bgaimana pelaporan
sy ttg kinerja kelurahan bulak? Apakah sdh d tindak lanjuti?']
```

Clear Username

```
dataclearusername_pos=[]
for line in dataclearhastag_pos:
    result=re.sub(r"@\S+","",line)
    dataclearusername_pos.append(result)

dataclearusername_neg=[]
for line in dataclearhastag_neg:
    result=re.sub(r"@\S+","",line)
    dataclearusername_neg.append(result)
```

```
[' Pasti untuk kebaikan warga kota sby, pemikiran bagi yg memahami
prosedur. dan pasti ada pemikiran yg lain nya', … '@SapawargaSby
Alhamdulillah,dengan adanya akun @SapawargaSby sangat membantu dan
respon cepat,terima kasih :)']
```

```
[' ada proses evakuasi bis mira yg terperosok ke kali, menyebabkan
macet panjang ', … '@SapawargaSby Slmt malam, bgaimana pelaporan
sy ttg kinerja kelurahan bulak? Apakah sdh d tindak lanjuti?']
```

Case folding

```
datalower_pos=[]
for line in dataclearusername_pos:
    a=line.lower()
    datalower_pos.append(a)

datalower_neg=[]
for line in dataclearusername_neg:
    a=line.lower()
    datalower_neg.append(a)
```

```
[' pasti untuk kebaikan warga kota sby, pemikiran bagi yg memahami
prosedur. dan pasti ada pemikiran yg lain nya', … '@sapawargasby
alhamdulillah,dengan adanya akun @sapawargasby sangat membantu dan
respon cepat,terima kasih :)']
```

```
[' ada proses evakuasi bis mira yg terperosok ke kali, menyebabkan
macet panjang ', … '@sapawargasby slmt malam, bgaimana pelaporan
sy ttg kinerja kelurahan bulak? apakah sdh d tindak lanjuti?']
```

Stemming

```
factory=StemmerFactory()
stemmer=factory.create_stemmer()
datastemmed_pos=map(lambda x: stemmer.stem(x), datalower_pos)
databersih_pos=map(lambda x: x.translate(str.maketrans('', '',
string.punctuation)), datastemmed_pos)
databersih_pos=list(databersih_pos)

factory=StemmerFactory()
stemmer=factory.create_stemmer()
datastemmed_neg=map(lambda x: stemmer.stem(x), datalower_neg)
databersih_neg=map(lambda x: x.translate(str.maketrans('', '',
string.punctuation)), datastemmed_neg)
databersih_neg=list(databersih_neg)
```

```
['pasti untuk baik warga kota sby pikir bagi yg paham prosedur dan
pasti ada pikir yg lain nya',… 'alhamdulillah dengan ada akun
sangat bantu dan respon cepat terima kasih']
```

```
['ada proses evakuasi bis mira yg perosok ke kali sebab macet
panjang',  …  'slmt malam bgaimana lapor sy ttg kerja lurah bulak
apakah sdh d tindak lanjut']
```

Stopwords

```
stopwords=open('stopwords.txt', 'r').read()

satudata_pos=[]
datafinal_pos=[]
df_pos=[]
for line in databersih_pos:
    wo_pos = word_tokenize(line)
    wo_pos = [word for word in wo_pos if not word in stopwords and
not word[0].isdigit()]
    datafinal_pos.append(wo_pos)
    df_pos.append(" ".join(wo_pos))
for l in datafinal_pos:
    satudata_pos+= l
final_pos={v: satudata_pos.count(v) for v in set(satudata_pos)}

satudata_neg=[]
datafinal_neg=[]
df_neg=[]
for line in databersih_neg:
    wo_neg = word_tokenize(line)
    wo_neg = [word for word in wo_neg if not word in stopwords and
not word[0].isdigit()]
    datafinal_neg.append(wo_neg)
    df_neg.append(" ".join(wo_neg))
for l in datafinal_neg:
    satudata_neg+= l
final_neg={v: satudata_neg.count(v) for v in set(satudata_neg)}
```

```
['untuk baik warga kota pikir paham prosedur pikir', … 'akun bantu
respon cepat']
```

```
['proses evakuasi mira perosok macet', … 'slmt malam bgaimana
lapor ttg kerja lurah bulak tindak lanjut']
```

Merubah Data str

```
a = str(df_pos)
positif =re.sub(r"'","",a)

b = str(df_neg)
negatif =re.sub(r"'","",b)
```

```
[untuk baik warga kota pikir paham prosedur pikir,… akun bantu
respon cepat]
```

```
[proses evakuasi mira perosok macet, … slmt malam bgaimana lapor
ttg kerja lurah bulak tindak lanjut]
```

Word Cloud

```
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt
%matplotlib inline
from subprocess import check_output
from Word Cloud import Word Cloud, STOPWORDS
mpl.rcParams['font.size']=12                #10
mpl.rcParams['savefig.dpi']=100             #72
mpl.rcParams['figure.subplot.bottom']=.1
```

Word Cloud Positif

```
Word Cloud = Word Cloud(collocations = True,
                        background_color='white',
                        stopwords=stopwords,
                        max_words=20,
                        max_font_size=200,
                        random_state=42
                       ).generate(positif)
print(Word Cloud)
fig = plt.figure(1)
plt.imshow(Word Cloud)
plt.axis('off')
plt.show()
fig.savefig("word1.png", dpi=900)
```

Word Cloud Negatif

```
Word Cloud = Word Cloud(collocations = False,
                        background_color='black',
                        stopwords=stopwords,
                        max_words=20,
                        max_font_size=200,
                        random_state=42
                       ).generate(negatif)
print(Word Cloud)
fig = plt.figure(1)
plt.imshow(Word Cloud)
plt.axis('off')
plt.show()
fig.savefig("word1.png", dpi=900)
```

Lampiran 7. Analisis Klasifikasi Setiap Metode

```
import numpy
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt
import imblearn
import optunity
import optunity.metrics
from sklearn.metrics import classification_report, confusion_matrix
from __future__ import print_function
from sklearn.model_selection import
train_test_split,cross_val_score,StratifiedKFold,KFold,ShuffleSplit,S
tratifiedShuffleSplit
from sklearn.feature_selection import SelectPercentile, f_classif
from sklearn.naive_bayes import BernoulliNB
from sklearn.svm import LinearSVC, SVC
from sklearn.Linear_model import LogisticRegression
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import classification_report
from sklearn.utils import shuffle
from sklearn import svm, datasets
from sklearn.metrics import roc_curve, auc
from scipy import interp
from imblearn.over_sampling import SMOTE
from sklearn import metrics
from ggplot import *
import sklearn.svm
```

Naïve Bayes Classifier

```
Y_new = DataFrame.as_matrix(Y_nya)
X_new, Y_new = X_, Y_new
kfold= StratifiedKFold(n_splits=10, shuffle=False)
kfold.get_n_splits(X_new)
kfold.get_n_splits(Y_new)
cl= BernoulliNB()
smote=SMOTE()

i=1
for train, test in kfold.split(X_new, Y_new):
    (X_new[train], X_new[test])
    (Y_new[train], Y_new[test])
    hitung= cl.fit(X_new[train], Y_new[train])
    predictNBC= cl.predict(X_new[test])
    fpr, tpr, _ = metrics.roc_curve(Y_new[test], predictNBC)
    auc_ = metrics.auc(fpr,tpr)

    X_trainsmote, Y_trainsmote =
smote.fit_sample(X_new[train],Y_new[train])
    hitungsmote= cl.fit(X_trainsmote, Y_trainsmote)
    predictNBCsmote= cl.predict(X_new[test])
```

```
    fprsm, tprsm, _ = metrics.roc_curve(Y_new[test], predictNBCsmote)
    aucsm_ = metrics.auc(fprsm,tprsm)
    print ("----------FOLD KE = {:.0f}----------".format(i))
    numpy.savetxt("D:/NBC/Xawal%s.csv"%i, X_new[train],
delimiter=",")
    numpy.savetxt("D:/NBC/Yawal%s.csv"%i, Y_new[train],
delimiter=",")
    numpy.savetxt("D:/NBC/Xsmote%s.csv"%i, X_trainsmote,
delimiter=",")
    numpy.savetxt("D:/NBC/Ysmote%s.csv"%i, Y_trainsmote,
delimiter=",")
    numpy.savetxt("D:/NBC/Xawaltest%s.csv"%i, X_new[test],
delimiter=",")
    numpy.savetxt("D:/NBC/Yawaltest%s.csv"%i, Y_new[test],
delimiter=",")
    i=i+1
    print ("NBC data AWAL")
    print (confusion_matrix(Y_new[test], predictNBC))
    print (classification_report(Y_new[test], predictNBC))
    print ()
    print ("NBC data SMOTE")
    print (confusion_matrix(Y_new[test], predictNBCsmote))
    print (classification_report(Y_new[test], predictNBCsmote))
    print
("============================================================")
    print ("Accuracy data awal =
{:.2f}".format(accuracy_score(Y_new[test], predictNBC)))
    print ("Accuracy data SMOTE =
{:.2f}".format(accuracy_score(Y_new[test], predictNBCsmote)))
    print ("Area Under Curve ROC data awal = {:.2f}".format(auc_))
    print ("Area Under Curve ROC data SMOTE = {:.2f}".format(aucsm_))
    print
("============================================================")
```

Support Vector Machine *Kernel Linear*

```
lne= SVC(C=100, kernel='Linear')
i=1
for train, test in kfold.split(X_new, Y_new):
    (X_new[train], X_new[test])
    (Y_new[train], Y_new[test])
    hitung= lne.fit(X_new[train], Y_new[train])
    predictLNE= lne.predict(X_new[test])
    fpr, tpr, _ = metrics.roc_curve(Y_new[test], predictLNE)
    auc_ = metrics.auc(fpr,tpr)

    X_trainsmote, Y_trainsmote =
smote.fit_sample(X_new[train],Y_new[train])
    hitungsmote= lne.fit(X_trainsmote, Y_trainsmote)
    predictLNEsmote= lne.predict(X_new[test])
    fprsm, tprsm, _ = metrics.roc_curve(Y_new[test], predictLNEsmote)
```

```
    aucsm_ = metrics.auc(fprsm,tprsm)
    print ("----------FOLD KE = {:.0f}----------".format(i))
    numpy.savetxt("D:/Linear100/Xawal%s.csv"%i, X_new[train],
delimiter=",")
    numpy.savetxt("D:/Linear100/Yawal%s.csv"%i, Y_new[train],
delimiter=",")
    numpy.savetxt("D:/Linear100/Xsmote%s.csv"%i, X_trainsmote,
delimiter=",")
    numpy.savetxt("D:/Linear100/Ysmote%s.csv"%i, Y_trainsmote,
delimiter=",")
    i=i+1
    print ("Linear data AWAL")
    print (confusion_matrix(Y_new[test], predictLNE))
    print (classification_report(Y_new[test], predictLNE))
    print ()
    print ("Linear data SMOTE")
    print (confusion_matrix(Y_new[test], predictLNEsmote))
    print (classification_report(Y_new[test], predictLNEsmote))
    print
("============================================================")
    print ("Accuracy data awal =
{:.2f}".format(accuracy_score(Y_new[test], predictLNE)))
    print ("Accuracy data SMOTE =
{:.2f}".format(accuracy_score(Y_new[test], predictLNEsmote)))
    print ("Area Under Curve ROC data awal = {:.2f}".format(auc_))
    print ("Area Under Curve ROC data SMOTE = {:.2f}".format(aucsm_))
    print
("============================================================")
```

Support Vector Machine *Kernel* RBF

```
rbf= SVC(C=100, gamma=0.1, kernel='rbf')
i=1
for train, test in kfold.split(X_new, Y_new):
    (X_new[train], X_new[test])
    (Y_new[train], Y_new[test])
    hitung= rbf.fit(X_new[train], Y_new[train])
    predictRBF= rbf.predict(X_new[test])
    fpr, tpr, _ = metrics.roc_curve(Y_new[test], predictRBF)
    auc_ = metrics.auc(fpr,tpr)

    X_trainsmote, Y_trainsmote =
smote.fit_sample(X_new[train],Y_new[train])
    hitungsmote= rbf.fit(X_trainsmote, Y_trainsmote)
    predictRBFsmote= rbf.predict(X_new[test])
    fprsm, tprsm, _ = metrics.roc_curve(Y_new[test], predictRBFsmote)
    aucsm_ = metrics.auc(fprsm,tprsm)
    print ("----------FOLD KE = {:.0f}----------".format(i))
    numpy.savetxt("D:/RBF100_01/Xawal%s.csv"%i, X_new[train],
delimiter=",")
    numpy.savetxt("D:/RBF100_01/Yawal%s.csv"%i, Y_new[train],
delimiter=",")
    numpy.savetxt("D:/RBF100_01/Xsmote%s.csv"%i, X_trainsmote,
```

```
delimiter=",")
    numpy.savetxt("D:/RBF100_01/Ysmote%s.csv"%i, Y_trainsmote,
delimiter=",")
    i=i+1
    print ("RBF data AWAL")
    print (confusion_matrix(Y_new[test], predictRBF))
    print (classification_report(Y_new[test], predictRBF))
    print ()
    print ("RBF data SMOTE")
    print (confusion_matrix(Y_new[test], predictRBFsmote))
    print (classification_report(Y_new[test], predictRBFsmote))
    print
("============================================================")
    print ("Accuracy data awal =
{:.2f}".format(accuracy_score(Y_new[test], predictRBF)))
    print ("Accuracy data SMOTE =
{:.2f}".format(accuracy_score(Y_new[test], predictRBFsmote)))
    print ("Area Under Curve ROC data awal = {:.2f}".format(auc_))
    print ("Area Under Curve ROC data SMOTE = {:.2f}".format(aucsm_))
    print
("============================================================")
```

AUC Regresi Logistik Data SMOTE

```
lr= LogisticRegression()
i=1
for train, test in kfold.split(X_new, Y_new):
    (X_new[train], X_new[test])
    (Y_new[train], Y_new[test])
    hitung= lr.fit(X_new[train], Y_new[train])
    predictLR= lr.predict(X_new[test])
    fpr, tpr, _ = metrics.roc_curve(Y_new[test], predictLR)
    auc_ = metrics.auc(fpr,tpr)

    X_trainsmote, Y_trainsmote =
smote.fit_sample(X_new[train],Y_new[train])
    hitungsmote= lr.fit(X_trainsmote, Y_trainsmote)
    predictLRsmote= lr.predict(X_new[test])
    fprsm, tprsm, _ = metrics.roc_curve(Y_new[test], predictLRsmote)
    aucsm_ = metrics.auc(fprsm,tprsm)
    print ("----------FOLD KE = {:.0f}----------".format(i))
    numpy.savetxt("D:/LR/Xawal%s.csv"%i, X_new[train], delimiter=",")
    numpy.savetxt("D:/LR/Yawal%s.csv"%i, Y_new[train], delimiter=",")
    numpy.savetxt("D:/LR/Xsmote%s.csv"%i, X_trainsmote,
delimiter=",")
    numpy.savetxt("D:/LR/Ysmote%s.csv"%i, Y_trainsmote,
delimiter=",")
    i=i+1
    print ("LR data AWAL")
    print (confusion_matrix(Y_new[test], predictLR))
    print (classification_report(Y_new[test], predictLR))
    print ()
    print ("LR data SMOTE")
```

```
    print (confusion_matrix(Y_new[test], predictLRsmote))
    print (classification_report(Y_new[test], predictLRsmote))
    print
("============================================================")
    print ("Accuracy data awal =
{:.2f}".format(accuracy_score(Y_new[test], predictLR)))
    print ("Accuracy data SMOTE =
{:.2f}".format(accuracy_score(Y_new[test], predictLRsmote)))
    print ("Area Under Curve ROC data awal = {:.2f}".format(auc_))
    print ("Area Under Curve ROC data SMOTE = {:.2f}".format(aucsm_))
    print
("============================================================")
```

Lampiran 8. Perhitungan Manual NBC Data Awal

Perhitungan Prior Positif
$P\left(V_{positif}\right)=\frac{\left\|\text{doc}_{positif}\right\|}{\left\|\text{training}\right\|}=\frac{37}{169}=0,219$
Perhitungan Prior Negatif
$P\left(V_{negatif}\right)=\frac{\left\|\text{doc}_{negatif}\right\|}{\left\|\text{training}\right\|}=\frac{132}{169}=0,781$
Perhitungan Posterior Positif
$P\left(a_1\|v_{pos}\right)=\frac{n_1+1}{\left\|n+\text{kosa kata}\right\|}=\frac{2+1}{2325+3689}=0.0005$ $P\left(a_2\|v_{pos}\right)=\frac{n_2+1}{\left\|n+\text{kosa kata}\right\|}=\frac{1+1}{2325+3689}=0.0003$ $\vdots$ $P\left(a_{3689}\|v_{pos}\right)=\frac{n_{3689}+1}{\left\|n+\text{kosa kata}\right\|}=\frac{4+1}{2325+3689}=0.0008$
Perhitungan Posterior Negatif
$P\left(a_1\|v_{pos}\right)=\frac{n_1+1}{\left\|n+\text{kosa kata}\right\|}=\frac{2+1}{10282+3689}=0.0002$ $P\left(a_2\|v_{pos}\right)=\frac{n_2+1}{\left\|n+\text{kosa kata}\right\|}=\frac{1+1}{10282+3689}=0.0001$ $\vdots$ $P\left(a_{3689}\|v_{pos}\right)=\frac{n_{3689}+1}{\left\|n+\text{kosa kata}\right\|}=\frac{4+1}{10282+3689}=0.0004$
Perhitungan *Tweet* Ke-i Sentimen Positif

$$P(v_{pos})\prod_{i=1}^{3689}P(a_i|v_j)=0,219(0,0005^0\times0,0003^0\times\ldots\times0,0008^0)=0,999$$

$$P(v_{pos})\prod_{i=1}^{3689}P(a_i|v_j)=0,219(0,0005^0\times0,0003^0\times\ldots\times0,0008^0)=0,183$$

$$\vdots$$

$$P(v_{pos})\prod_{i=1}^{3689}P(a_i|v_j)=0,219(0,0005^0\times0,0003^0\times\ldots\times0,0008^0)=0,967$$

Perhitungan *Tweet* Ke-i Sentimen Negatif

$$P(v_{pos})\prod_{i=1}^{3689}P(a_i|v_j)=0,781(0,0002^0\times0,0001^0\times\ldots\times0,0004^0)=0,001$$

$$P(v_{pos})\prod_{i=1}^{3689}P(a_i|v_j)=0,781(0,0002^0\times0,0001^0\times\ldots\times0,0004^0)=0,817$$

$$\vdots$$

$$P(v_{pos})\prod_{i=1}^{3689}P(a_i|v_j)=0,781(0,0002^0\times0,0001^0\times\ldots\times0,0004^0)=0,033$$

Lampiran 9. Rangkuman Performa Klasifikasi Data Awal Metode SVM RBF

C	*Gamma*	*Subset*	*Accuracy*	*Precision*	*Recall*	AUC
0.01	0.01	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
	0.1	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50

		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
	1	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
	10	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
	100	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
0.1	0.01	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50

		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
	0.1	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
	1	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
	10	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
	100	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50

		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
1	0.01	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.00	0.00	0.50
	0.1	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.79	1.00	0.03	0.51
		Subset ke-9	0.78	0.00	0.00	0.50
		Subset ke-10	0.78	0.00	0.00	0.50
	Rata-rata		0.78	0.10	0.00	0.50
	1	*Subset* ke-1	0.79	1.00	0.03	0.51
		Subset ke-2	0.79	1.00	0.03	0.51
		Subset ke-3	0.80	1.00	0.14	0.54
		Subset ke-4	0.79	1.00	0.05	0.53
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.74	0.29	0.14	0.52
		Subset ke-7	0.79	1.00	0.05	0.53
		Subset ke-8	0.79	1.00	0.05	0.53
		Subset ke-9	0.80	1.00	0.08	0.54
		Subset ke-10	0.81	0.83	0.14	0.57

	Rata-rata		0.79	0.91	0.07	0.53
	10	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.72	0.00	0.00	0.46
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.11	0.55
		Subset ke-10	0.81	1.00	0.11	0.56
	Rata-rata		0.78	0.30	0.03	0.51
	100	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.72	0.00	0.00	0.46
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.11	0.55
		Subset ke-10	0.81	1.00	0.11	0.56
	Rata-rata		0.78	0.20	0.02	0.51
100	0.01	*Subset* ke-1	0.79	0.56	0.27	0.60
		Subset ke-2	0.83	0.58	0.41	0.68
		Subset ke-3	0.83	0.76	0.35	0.66
		Subset ke-4	0.81	0.60	0.41	0.66
		Subset ke-5	0.79	0.55	0.16	0.56
		Subset ke-6	0.80	0.34	0.41	0.59
		Subset ke-7	0.80	0.61	0.30	0.62
		Subset ke-8	0.81	0.62	0.35	0.65
		Subset ke-9	0.82	0.77	0.27	0.62
		Subset ke-10	0.82	0.60	0.50	0.70
	Rata-rata		0.81	0.60	0.34	0.63
	0.1	*Subset* ke-1	0.79	0.52	0.32	0.62
		Subset ke-2	0.80	0.57	0.32	0.65
		Subset ke-3	0.81	0.64	0.38	0.66
		Subset ke-4	0.79	0.48	0.35	0.67
		Subset ke-5	0.78	0.45	0.14	0.55
		Subset ke-6	0.67	0.33	0.41	0.58
		Subset ke-7	0.79	0.43	0.16	0.60

		Subset ke-8	0.84	0.71	0.32	0.70
		Subset ke-9	0.79	0.56	0.38	0.64
		Subset ke-10	0.81	0.53	0.47	0.72
	Rata-rata		0.79	0.52	0.33	0.64
	1	*Subset* ke-1	0.80	0.60	0.08	0.56
		Subset ke-2	0.81	1.00	0.11	0.58
		Subset ke-3	0.81	1.00	0.22	0.60
		Subset ke-4	0.81	1.00	0.16	0.58
		Subset ke-5	0.80	1.00	0.05	0.54
		Subset ke-6	0.74	0.32	0.19	0.57
		Subset ke-7	0.80	1.00	0.05	0.56
		Subset ke-8	0.81	1.00	0.16	0.58
		Subset ke-9	0.85	1.00	0.32	0.66
		Subset ke-10	0.84	0.86	0.33	0.69
	Rata-rata		0.81	0.88	0.17	0.59
	10	*Subset* ke-1	0.80	0.60	0.08	0.57
		Subset ke-2	0.81	1.00	0.11	0.58
		Subset ke-3	0.82	1.00	0.22	0.62
		Subset ke-4	0.81	1.00	0.16	0.58
		Subset ke-5	0.80	1.00	0.05	0.54
		Subset ke-6	0.74	0.32	0.19	0.57
		Subset ke-7	0.80	1.00	0.05	0.58
		Subset ke-8	0.81	1.00	0.16	0.58
		Subset ke-9	0.85	1.00	0.32	0.66
		Subset ke-10	0.84	0.86	0.33	0.69
	Rata-rata		0.81	0.88	0.17	0.60
	100	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.72	0.00	0.00	0.46
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.11	0.55
		Subset ke-10	0.81	1.00	0.11	0.56
	Rata-rata		0.78	0.30	0.03	0.51

Lampiran 10. Rangkuman Performa Klasifikasi Data SMOTE Metode SVM RBF

C	*Gamma*	*Subset 10-Folds*	*Accuracy*	*Precision*	*Recall*	AUC
0.01	0.01	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.82	0.89	0.22	0.60
		Subset ke-4	0.79	1.00	0.03	0.51
		Subset ke-5	0.79	1.00	0.50	0.53
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.79	1.00	0.50	0.53
		Subset ke-8	0.85	0.92	0.32	0.66
		Subset ke-9	0.85	1.00	0.03	0.65
		Subset ke-10	0.80	0.67	0.11	0.55
	Rata-rata		0.80	0.65	0.17	0.55
	0.1	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.77	0.00	0.00	0.49
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.79	1.00	0.03	0.51
		Subset ke-9	0.82	0.10	0.16	0.58
		Subset ke-10	0.81	1.00	0.14	0.57
	Rata-rata		0.79	0.31	0.04	0.52
	1	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.77	0.00	0.00	0.49
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.08	0.54
		Subset ke-10	0.80	1.00	0.08	0.54
	Rata-rata		0.78	0.30	0.02	0.51
	10	*Subset* ke-1	0.78	0.00	0.00	0.50

		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.77	0.00	0.00	0.49
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.08	0.54
		Subset ke-10	0.80	1.00	0.08	0.54
	Rata-rata		0.78	0.30	0.02	0.51
	100	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.77	0.00	0.00	0.49
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.08	0.54
		Subset ke-10	0.80	1.00	0.08	0.54
	Rata-rata		0.78	0.30	0.02	0.51
0.1	0.01	*Subset* ke-1	0.80	0.70	0.19	0.58
		Subset ke-2	0.79	1.00	0.03	0.51
		Subset ke-3	0.80	1.00	0.11	0.55
		Subset ke-4	0.79	1.00	0.05	0.53
		Subset ke-5	0.79	1.00	0.05	0.53
		Subset ke-6	0.75	0.33	0.16	0.54
		Subset ke-7	0.79	1.00	0.05	0.53
		Subset ke-8	0.79	1.00	0.05	0.53
		Subset ke-9	0.85	1.00	0.30	0.65
		Subset ke-10	0.82	0.75	0.25	0.61
	Rata-rata		0.80	0.88	0.12	0.56
	0.1	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.79	1.00	0.03	0.51

		Subset ke-7	0.79	1.00	0.03	0.51
		Subset ke-8	0.79	1.00	0.03	0.51
		Subset ke-9	0.82	0.10	0.16	0.58
		Subset ke-10	0.81	1.00	0.14	0.57
	Rata-rata		0.79	0.51	0.04	0.52
	1	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.77	0.00	0.00	0.49
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.08	0.54
		Subset ke-10	0.81	1.00	0.11	0.56
	Rata-rata		0.79	0.30	0.02	0.51
	10	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.77	0.00	0.00	0.49
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.08	0.54
		Subset ke-10	0.80	1.00	0.08	0.54
	Rata-rata		0.78	0.30	0.02	0.51
	100	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.77	0.00	0.00	0.49
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.08	0.54
		Subset ke-10	0.80	1.00	0.08	0.54
	Rata-rata		0.78	0.30	0.02	0.51

1	0.01	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.79	1.00	0.03	0.51
		Subset ke-4	0.80	1.00	0.08	0.54
		Subset ke-5	0.78	0.00	0.00	0.50
		Subset ke-6	0.78	0.00	0.00	0.50
		Subset ke-7	0.78	0.79	1.00	0.05
		Subset ke-8	0.80	1.00	0.08	0.54
		Subset ke-9	0.81	1.00	0.14	0.57
		Subset ke-10	0.80	0.71	0.14	0.56
	Rata-rata		0.79	0.55	0.15	0.48
	0.1	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.79	1.00	0.03	0.51
		Subset ke-4	0.79	1.00	0.03	0.51
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.73	0.00	0.00	0.47
		Subset ke-7	0.79	1.00	0.05	0.53
		Subset ke-8	0.79	1.00	0.03	0.51
		Subset ke-9	0.82	1.00	0.19	0.59
		Subset ke-10	0.82	1.00	0.17	0.58
	Rata-rata		0.79	0.70	0.05	0.52
	1	*Subset* ke-1	0.79	1.00	0.05	0.53
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.79	1.00	0.03	0.51
		Subset ke-4	0.79	1.00	0.03	0.51
		Subset ke-5	0.79	1.00	0.05	0.53
		Subset ke-6	0.79	1.00	0.03	0.51
		Subset ke-7	0.79	1.00	0.05	0.53
		Subset ke-8	0.79	1.00	0.05	0.53
		Subset ke-9	0.82	1.00	0.19	0.59
		Subset ke-10	0.85	0.92	0.30	0.66
	Rata-rata		0.80	0.89	0.08	0.54
	10	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51

		Subset ke-6	0.72	0.00	0.00	0.46
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.11	0.55
		Subset ke-10	0.81	1.00	0.11	0.56
	Rata-rata		0.78	0.30	0.03	0.51
	100	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.72	0.00	0.00	0.46
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.11	0.55
		Subset ke-10	0.81	1.00	0.11	0.56
	Rata-rata		0.78	0.30	0.03	0.51
10	0.01	*Subset* ke-1	0.79	0.56	0.27	0.60
		Subset ke-2	0.83	0.58	0.41	0.68
		Subset ke-3	0.83	0.76	0.35	0.66
		Subset ke-4	0.81	0.60	0.41	0.66
		Subset ke-5	0.79	0.55	0.16	0.56
		Subset ke-6	0.80	0.34	0.41	0.59
		Subset ke-7	0.80	0.61	0.30	0.62
		Subset ke-8	0.81	0.62	0.35	0.65
		Subset ke-9	0.82	0.77	0.27	0.62
		Subset ke-10	0.82	0.60	0.50	0.70
	Rata-rata		0.81	0.60	0.34	0.63
	0.1	*Subset* ke-1	0.79	0.52	0.32	0.62
		Subset ke-2	0.80	0.57	0.32	0.65
		Subset ke-3	0.81	0.64	0.38	0.66
		Subset ke-4	0.79	0.48	0.35	0.67
		Subset ke-5	0.78	0.45	0.14	0.55
		Subset ke-6	0.67	0.33	0.41	0.58
		Subset ke-7	0.79	0.43	0.16	0.60
		Subset ke-8	0.84	0.71	0.32	0.70
		Subset ke-9	0.79	0.56	0.38	0.64
		Subset ke-10	0.81	0.53	0.47	0.72

	Rata-rata		0.79	0.52	0.33	0.64
	1	*Subset* ke-1	0.80	0.60	0.08	0.56
		Subset ke-2	0.81	1.00	0.11	0.58
		Subset ke-3	0.81	1.00	0.22	0.60
		Subset ke-4	0.81	1.00	0.16	0.58
		Subset ke-5	0.80	1.00	0.05	0.54
		Subset ke-6	0.74	0.32	0.19	0.57
		Subset ke-7	0.80	1.00	0.05	0.56
		Subset ke-8	0.81	1.00	0.16	0.58
		Subset ke-9	0.85	1.00	0.32	0.66
		Subset ke-10	0.84	0.86	0.33	0.69
	Rata-rata		0.81	0.88	0.17	0.59
	10	*Subset* ke-1	0.80	0.60	0.08	0.57
		Subset ke-2	0.81	1.00	0.11	0.58
		Subset ke-3	0.82	1.00	0.22	0.62
		Subset ke-4	0.81	1.00	0.16	0.58
		Subset ke-5	0.80	1.00	0.05	0.54
		Subset ke-6	0.74	0.32	0.19	0.57
		Subset ke-7	0.80	1.00	0.05	0.58
		Subset ke-8	0.81	1.00	0.16	0.58
		Subset ke-9	0.85	1.00	0.32	0.66
		Subset ke-10	0.84	0.86	0.33	0.69
	Rata-rata		0.81	0.88	0.17	0.60
	100	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.72	0.00	0.00	0.46
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.11	0.55
		Subset ke-10	0.81	1.00	0.11	0.56
	Rata-rata		0.78	0.30	0.03	0.51
100	0.01	*Subset* ke-1	0.78	0.50	0.32	0.62
		Subset ke-2	0.81	0.58	0.49	0.69
		Subset ke-3	0.79	0.54	0.35	0.63
		Subset ke-4	0.79	0.52	0.46	0.67

		Subset ke-5	0.79	0.56	0.24	0.60
		Subset ke-6	0.67	0.33	0.51	0.61
		Subset ke-7	0.79	0.40	0.22	0.56
		Subset ke-8	0.79	0.52	0.35	0.63
		Subset ke-9	0.80	0.57	0.43	0.67
		Subset ke-10	0.83	0.60	0.58	0.74
	Rata-rata		0.78	0.51	0.40	0.64
	0.1	*Subset* ke-1	0.79	0.53	0.27	0.60
		Subset ke-2	0.78	0.47	0.24	0.58
		Subset ke-3	0.82	0.64	0.38	0.66
		Subset ke-4	0.77	0.46	0.35	0.62
		Subset ke-5	0.78	0.50	0.16	0.56
		Subset ke-6	0.69	0.33	0.41	0.59
		Subset ke-7	0.77	0.43	0.16	0.55
		Subset ke-8	0.82	0.67	0.32	0.64
		Subset ke-9	0.80	0.61	0.38	0.65
		Subset ke-10	0.80	0.55	0.47	0.68
	Rata-rata		0.78	0.52	0.31	0.61
	1	*Subset* ke-1	0.79	0.60	0.08	0.53
		Subset ke-2	0.80	1.00	0.11	0.55
		Subset ke-3	0.83	1.00	0.22	0.61
		Subset ke-4	0.81	1.00	0.14	0.57
		Subset ke-5	0.79	1.00	0.05	0.53
		Subset ke-6	0.73	0.32	0.19	0.54
		Subset ke-7	0.79	1.00	0.05	0.53
		Subset ke-8	0.81	1.00	0.14	0.57
		Subset ke-9	0.85	1.00	0.30	0.65
		Subset ke-10	0.84	0.86	0.33	0.66
	Rata-rata		0.80	0.88	0.16	0.57
	10	*Subset* ke-1	0.79	0.60	0.08	0.53
		Subset ke-2	0.80	1.00	0.11	0.55
		Subset ke-3	0.83	1.00	0.22	0.61
		Subset ke-4	0.81	1.00	0.14	0.57
		Subset ke-5	0.79	1.00	0.05	0.53
		Subset ke-6	0.73	0.32	0.19	0.54
		Subset ke-7	0.79	1.00	0.05	0.53
		Subset ke-8	0.81	1.00	0.14	0.57
		Subset ke-9	0.86	1.00	0.35	0.68

		Subset ke-10	0.84	0.86	0.33	0.66
	Rata-rata		0.81	0.88	0.17	0.58
	100	*Subset* ke-1	0.78	0.00	0.00	0.50
		Subset ke-2	0.78	0.00	0.00	0.50
		Subset ke-3	0.78	0.00	0.00	0.50
		Subset ke-4	0.78	0.00	0.00	0.50
		Subset ke-5	0.79	1.00	0.03	0.51
		Subset ke-6	0.72	0.00	0.00	0.46
		Subset ke-7	0.78	0.00	0.00	0.50
		Subset ke-8	0.78	0.00	0.00	0.50
		Subset ke-9	0.80	1.00	0.11	0.55
		Subset ke-10	0.81	1.00	0.11	0.56
	Rata-rata		0.78	0.30	0.03	0.51

Lampiran 11. *Output* Persamaan *Hyperplane*

Data Awal

```
-       2.1386 * < 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 … 0
0 0 0> * X]
+       4.3553 * < 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 … 0
0 0 0> * X]
                          ⋮
-       0.23   * < 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 … 0
0 0 0> * X]
-       3.0906
```

Data SMOTE

```
-       0.0051 * < 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 … 0
0 0 0> * X]
-       1.4195 * < 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 … 0
0 0 0> * X]
                          ⋮
+       2.962  * < 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 … 0
0 0 0> * X]
-       2.9316
```

Lampiran 12. *Output* Model Regresi Logistik Data Awal

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-8.69E+34	4.18E+10	-2.08E+24	0
xaamiin	-2.05E+43	2.52E+25	-8.16E+17	0
xabdi	1.42E+34	2.55E+12	5.57E+21	0
xacara	2.98E+36	5.46E+21	5.46E+14	0
xachmad	-3.72E+39	1.72E+24	-2.17E+15	0
xacung	-1.17E+40	8.35E+24	-1.40E+15	0
xadakan	4.34E+40	9.47E+24	4.58E+15	0
xadik	-3.25E+40	1.62E+25	-2.01E+15	0
xadmin	1.07E+35	3.40E+11	3.14E+23	0
xadminitrasi	-4.70E+40	3.96E+25	-1.19E+15	0
xadong2	-2.87E+39	1.38E+25	-2.08E+14	0
xagung	-8.46E+40	1.57E+25	-5.38E+15	0
xagus	3.59E+38	2.27E+25	1.58E+13	0
xagustus	6.54E+38	3.55E+24	1.84E+14	0
⋮	⋮	⋮	⋮	⋮
xsupoltas	-1.11E+39	1.55E+25	-7.16E+13	0
xsuwandhie	-3.15E+50	9.42E+38	-3.34E+11	0
xsuwun	-1.17E+27	8.43E+11	-1.39E+15	0
xtaat	6.38E+39	1.45E+25	4.40E+14	0
xtanda	1.13E+40	4.42E+25	2.56E+14	0
xtanggap	3.76E+21	1.05E+08	3.57E+13	0
xtarip	-6.59E+24	1.32E+11	-4.98E+13	0
xtempat	-2.02E+38	6.62E+24	-3.05E+13	0
xterimakasih	-1.45E+24	2.81E+08	-5.17E+15	0
xterimakasiih	4.78E+21	99010066	4.83E+13	0
xthx	3.20E+26	6.34E+11	5.05E+14	0
xtidak	-1.59E+25	3.85E+10	-4.12E+14	0
xtiket	2.89E+39	6.00E+24	4.81E+14	0
xtipu	3.95E+39	9.45E+24	4.18E+14	0
xtkasih	-3.20E+20	1.06E+08	-3.02E+12	0
xtkp	1.08E+40	2.66E+25	4.07E+14	0
xtrailer	8.58E+39	2.30E+25	3.74E+14	0

Lampiran 13. *Output* Model Regresi Logistik Data SMOTE

	Estimate	Std. Error	z value	Pr(>|z|)
xaamiin	-1.39E+29	1.15E+21	-1.2E+08	0
xabai	9.02E+28	2.39E+21	37643916	0
xabdi	-1.66E+25	42329402	-3.9E+17	0
xabu2	2.43E+27	4.84E+11	5.01E+15	0
xacara	9.16E+25	1.16E+08	7.87E+17	0
xachmad	-8.51E+26	9.66E+10	-8.8E+15	0
xacung	5.54E+27	1.01E+12	5.51E+15	0
xadab	1.94E+29	5.32E+21	36523533	0
xadakan	-4.55E+27	2.87E+20	-1.6E+07	0
xadidas	-2.41E+29	8.62E+20	-2.8E+08	0
xadmin	-2.66E+25	23556530	-1.1E+18	0
xadminitrasi	-6.14E+27	1.92E+20	-3.2E+07	0
xadong2	3.99E+28	2.87E+20	1.39E+08	0
xagung	-3.80E+28	7.66E+20	-5E+07	0
xagus	-1.43E+27	3.73E+11	-3.8E+15	0
xagustus	-1.13E+27	6.86E+10	-1.6E+16	0
xahmad	-6.87E+28	5.75E+20	-1.2E+08	0
xair	1.53E+27	1.53E+11	9.99E+15	0
⋮	⋮	⋮	⋮	⋮
xsiyap	1.87E+15	54752917	34223738	0
xslip	-5.9E+16	1.16E+08	-5.1E+08	0
xsungai	-2.8E+18	5.14E+10	-5.5E+07	0
xsungguh	2E+18	7.18E+10	27801317	0
xsuprijadi	-3.5E+18	7.57E+10	-4.7E+07	0
xsuruh	-3.4E+18	1.42E+11	-2.4E+07	0
xsuwun	-3.1E+15	21413401	-1.4E+08	0
xtarip	-3.9E+16	67412194	-5.8E+08	0
xteman	2.75E+18	4.88E+10	56230464	0
xterimakasih	-3.8E+14	18532173	-2.1E+07	0
xthx	7.32E+15	27528458	2.66E+08	0
xtidak	2.88E+15	57143439	50444610	0
xtrosobo	2.33E+28	8.62E+20	27030607	0
xuntuk	-1.5E+14	61248994	-2381289	0
xurus	7.38E+18	7.16E+10	1.03E+08	0
xwarga	-2.9E+18	7.24E+10	-4E+07	0
xwhatsapp	3.64E+18	3.96E+10	91968285	0

Lampiran 14. Surat Pernyataan Sumber Data

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini, mahasiswa Departemen Statistika FMKSD ITS :

Nama : Rakhmah Wahyu Mayasari

NRP : 062116 4500 0034

menyatakan bahwa data yang digunakan dalam Tugas Akhir / ~~Thesis~~ ini merupakan data sekunder yang diambil dari ~~Penelitian / Buku / Tugas Akhir / Thesis /~~ Publikasi lainnya yaitu:

Sumber : Twitter API (*Application Program Interface*)

Keterangan : Data *tweet* dengan *keywords* "SapawargaSby" dan "e100ss"

Surat Pernyataan ini dibuat dengan sebenarnya. Apabila terdapat pemalsuan data maka saya siap menerima sanksi sesuai aturan yang berlaku.

Mengetahui
Pembimbing Tugas Akhir

Dr. Dra. Kartika Fithriasari, M.Si
NIP. 19691212 199303 2 002

Surabaya, 26 Juli 2018

Rakhmah Wahyu Mayasari
NRP. 062116 4500 0034

*(coret yang tidak perlu)

(Halaman ini sengaja dikosongkan)

BIODATA PENULIS

Penulis memiliki nama lengkap Rakhmah Wahyu Mayasari, biasa disapa Maya, merupakan anak terakhir dari empat bersaudara. Penulis dilahirkan di Tulungagung pada tanggal 2 Mei 1995. Pendidikan formal yang pernah ditempuh penulis adalah SD Negeri 1 Karangrejo, SMP Negeri 1 Tulungagung, SMA Negeri 1 Kedungwaru, DIII Statistika ITS pada tahun 2016 diterima menjadi mahasiswa Jurusan Statistika ITS Program Studi Lintas Jalur. Selama menjadi Mahasiswa, penulis aktif dalam beberapa kegiatan kemahasiswaan di ITS, diantaranya menjadi anggota UKM PSM ITS pada tahun 2014, Staff Hubungan Luar UKM PSM ITS 2014/2015, dan Bendahara I UKM PSM ITS 2015/2016. Selain itu selama menjadi mahasiswa penulis juga berkesempatan magang di PT. Kelola Mina Laut Gresik di bagian *Quality Control* (QC) dan di Balai Penelitian Jeruk dan Buah Sub Tropika Batu, Malang. Penulis memiliki hobi menyanyi hingga mengantarkan UKM PSM ITS menjuarai di kancah nasional maupun intenasional. Informasi dan komunikasi lebih lanjut dengan penulis dapat menghubungi :

Email : wahyumaya00700@gmail.com
IDLine : wahyu_maya
Phone : 0857755234401

(Halaman ini sengaja dikosongkan)