ANALISIS SENTIMEN PUBLIK TERHADAP BJORKA DALAM INSIDEN KEBOCORAN DATA KOMINFO MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE

TUGAS AKHIR

Program Studi
S1 SISTEM INFORMASI

Oleh:
Rayhan Sabian
17410100144

FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA
2023

ANALISIS SENTIMEN PUBLIK TERHADAP BJORKA DALAM INSIDEN KEBOCORAN DATA KOMINFO MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk menyelesaikan
Program Sarjana

Oleh:

Nama	**: Rayhan Sabian**
NIM	**: 17410100144**
Program Studi	**: S1 Sistem Informasi**

FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA
2023

Tugas Akhir

ANALISIS SENTIMEN PUBLIK TERHADAP BJORKA DALAM INSIDEN KEBOCORAN DATA KOMINFO MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE

Dipersiapkan dan disusun oleh

Rayhan Sabian

NIM: 17410100144

Telah diperiksa, dibahas dan disetujui oleh Dewan Pembahas

Pada: 2 Januari 2023

Susunan Dewan Pembahas

Pembimbing

I. Dr. Drs. Antok Supriyanto, M.MT.
NIDN. 0726106201

II. Sulistiowati, S.Si., M.M.
NIDN. 0719016801

Pembahas

Dr. Eng. Valentinus Roby Hananto, S.Kom., M.Sc.
NIDN. 0715028903

Tugas Akhir ini telah diterima sebagai salah satu persyaratan, untuk memperoleh gelar Sarjana

Digitally signed by Universitas Dinamika
Date: 2023.01.25 10:03:31 +07'00'

Tri Sagirani, S.Kom., M.MT.

NIDN. 0731057301

Dekan Fakultas Teknologi dan Informatika

UNIVERSITAS DINAMIKA

"Everyone can achieve success, just some journeys are longer than others"

Gary Ongko Putra

Kupersembahkan ke Ibunda dan Ayah tercinta,

serta semua orang yang menyayangiku

SURAT PERNYATAAN

PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa Universitas Dinamika, Saya :

Nama	: Rayhan Sabian
NIM	: 17410100144
Program Studi	: S1 Sistem Informasi
Fakultas	: Fakultas Teknologi dan Informatika
Jenis Karya	: Tugas Akhir
Judul Karya	: **ANALISIS SENTIMEN PUBLIK TERHADAP BJORKA DALAM INSIDEN KEBOCORAN DATA KOMINFO MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE**

Menyatakan dengan sesungguhnya bahwa :

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, Saya menyetujui memberikan kepada **Universitas Dinamika** Hak Bebas Royalti Non-Eksklusif (***Non-Exclusive Royalty Free Right***) atas seluruh isi/sebagian karya ilmiah Saya tersebut diatas untuk disimpan, dialihmediakan, dan dikelola dalam bentuk pangkalan data (***database***) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama Saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta.
2. Karya tersebut diatas adalah hasil karya asli Saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya, atau pendapat orang lain yang ada dalam karya ilmiah ini semata-mata hanya sebagai rujukan yang dicantumkan dalam Daftar Pustaka Saya.
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiasi pada karya ilmiah ini, maka Saya bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada Saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, 7 Desember 2022

Rayhan Sabian
NIM : 17410100144

ABSTRAK

Akun dengan nama Bjorka mengklaim telah memperoleh miliaran data pendaftaran kartu SIM berupa Nomor Induk Kependudukan dan Kartu Keluarga dari *database* badan pemerintahan Kementerian Komunikasi dan Informatika (Kemkominfo), sehingga keamanan siber Kemkominfo pun dipertanyakan. Kemunculan hacker Bjorka ini menimbulkan berbagai tanggapan di Twitter, beberapa ada yang mendukung aksi Bjorka dan ada yang tidak setuju dengan aksi Bjorka. Maka diperlukannya analisis sentimen untuk mengetahui sentimen masyarakat lebih ke arah negatif atau positif, agar pemerintahan dapat melakukan evaluasi maupun rencana strategis pemerintah dalam menangani insiden kebocoran data ke depannya. Maka penelitian ini menggunakan *tweet* yang berisi tanggapan masyarakat untuk dilakukan prediksi sentimen negatif atau positif menggunakan algoritma *Support Vector machine*. Dari total 1017 data tanggapan masyarakat terhadap kebocoran data oleh Bjorka ditemukan 97.35% (990 *tweet*) memiliki sentimen negatif dan 2.65% (27 *tweet*) memiliki sentimen positif, sehingga dapat diketahui tanggapan publik lebih banyak beranggapan negatif terhadap kebocoran data yang dilakukan oleh Bjorka. Dari hasil sentimen tersebut juga menyatakan bahwa edukasi ke masyarakat terhadap kebocoran data yang dilakukan oleh Bjorka tidak terlalu krusial, pemerintahan bisa lebih fokus untuk menangani sektor yang lain seperti meningkatkan keamanan data itu sendiri maupun menyiapkan edukasi kebocoran data lainnya.

Kata Kunci: *support vector machine, kebocoran data, analisis sentimen*

KATA PENGANTAR

Puji syukur ke hadirat Allah SWT Tuhan Yang Maha Esa karena atas rahmat dan karunia-Nya, penulis dapat menyelesaikan Tugas Akhir yang berjudul "Analisis Sentimen Publik terhadap Bjorka dalam Insiden Kebocoran Data KOMINFO menggunakan Algoritma Support Vector Machine" dengan lancar meskipun penulis menyadari masih terdapat banyak kekurangan dalam laporan ini.

Penyelesaian Tugas Akhir ini tidak terlepas dari bantuan berbagai pihak yang telah memberikan banyak masukan, nasihat, saran, kritik dan dukungan moral maupun dukungan secara materil kepada penulis. Oleh karena itu penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu penyelesaian Laporan Tugas Akhir ini, kepada yang terhormat:

1. Bapak Prof. Dr. Budi Jatmiko, M.Pd. selaku Rektor Universitas Dinamika Surabaya.
2. Bapak Dr. Anjik Sukmaaji, S.Kom., M.Eng. selaku Ketua Program Studi S1 Sistem Informasi Universitas Dinamika.
3. Bapak Dr. Drs. Antok Supriyanto, M.MT. selaku dosen pembimbing yang telah memberikan dukungan penuh berupa motivasi, saran dan wawasan bagi penulis selama pelaksanaan tugas akhir dan pembuatan laporan tugas akhir.
4. Ibu Sulistiowati, S.Si, M.M. selaku dosen pembimbing yang telah memberikan dukungan penuh berupa motivasi, saran dan wawasan bagi penulis selama pelaksanaan tugas akhir dan pembuatan laporan tugas akhir.
5. Bapak Dr. Eng. Valentinus Roby Hananto, S.Kom., M.Sc. selaku dosen pembahas yang telah menyempurnakan tugas akhir ini.
6. Orang tua tercinta serta keluarga yang selalu mendoakan, mendukung dan memberikan semangat di setiap langkah dan aktivitas penulis.
7. Teman-teman tercinta yang memberikan motivasi penulis dalam menyelesaikan laporan tugas akhir ini.
8. Semua pihak yang tidak dapat disebutkan satu persatu dalam kesempatan ini.

Semoga Tuhan Yang Maha Esa memberikan imbalan yang setimpal atas segala bantuan yang telah diberikan. Penulis menyadari di dalam laporan tugas akhir ini masih memiliki banyak kekurangan, semoga laporan tugas akhir ini dapat bermanfaat bagi semua pihak dan dapat menjadi bahan acuan untuk penelitian selanjutnya.

Surabaya, 7 Desember 2022

Rayhan Sabian

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR LAMPIRAN

BAB I
PENDAHULUAN

1.1 Latar Belakang

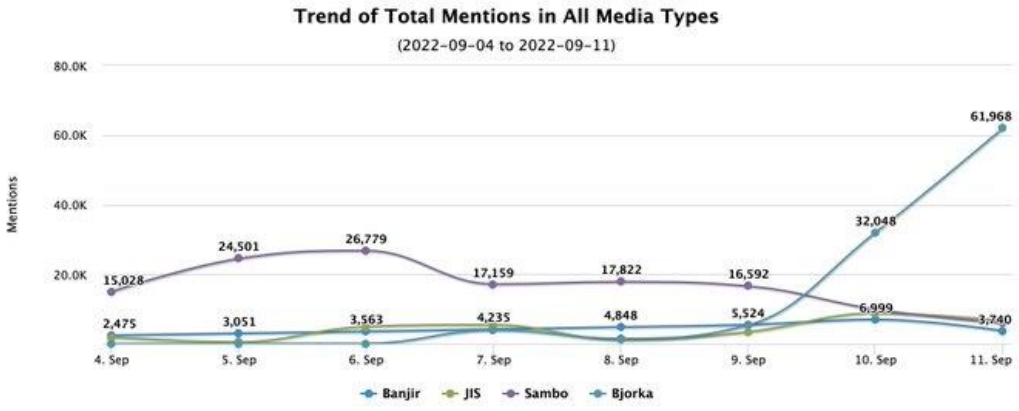
Data adalah catatan atas kumpulan fakta sehingga dapat dipahami bahwa data merupakan pernyataan yang diterima secara kenyataannya (diterima secara apa adanya). Pernyataan ini adalah hasil pengukuran atau pengamatan suatu variabel yang bentuknya dapat berupa angka, kata-kata, atau citra. Dalam keilmuan, fakta dikumpulkan untuk menjadi data. Data tersebut kemudian diolah sehingga dapat secara jelas dan tepat dimengerti oleh orang lain yang tidak langsung mengalaminya sendiri (Jonizar, 2020). Penggunaan teknologi informasi tidak jauh dari pengolahan maupun pemanfaatan suatu data sehingga untuk menjaga data tersebut dari terjadinya penyalahgunaan, dibutuhkan keamanan data dalam sebuah perangkat. Keamanan data merupakan sebuah prosedur dengan dukungan dari regulasi dan teknologi untuk melindungi data dari kerusakan data, modifikasi data, serta penyebaran data baik yang disengaja maupun tidak (Aprillia, 2022). Oleh sebab itu keamanan data merupakan tindakan yang perlu dilakukan oleh suatu perusahaan atau individu untuk melindungi ekosistem teknologi informasi, karena data merupakan dokumen penting yang terkadang bersifat rahasia sehingga jika terjadi pencurian data dapat merugikan perusahaan atau individu tersebut.

Kementerian Komunikasi dan Informatika (Kemkominfo) merupakan kementerian Indonesia yang bertugas dalam urusan dibidang komunikasi dan informatika. Kementerian Komunikasi dan Informatika sempat mengalami perubahan nama beberapakali, yaitu Departemen Penerangan (1945-1999), Kementerian Negara Komunikasi dan Informasi (2001-2005), dan Departemen Komunikasi dan Informatika (2005-2009) (Kementerian Komunikasi dan Informatika Republik indonesia, n.d.). Pada 30 Agustus 2022 “Miliaran data pendaftaran kartu SIM atau SIM card berupa nomor induk kependudukan (NIK) dan Kartu Keluarga (KK) diduga bocor di forum hacker. Keamanan siber Kementerian Komunikasi dan Informatika pun dipertanyakan. Akun dengan nama

Bjorka mengklaim memperoleh data tersebut dari data base Kementerian Komunikasi dan Informatika (Kemenkominfo).” (CNN Indonesia, 2022). Namun di saat yang sama, perwakilan Kementerian Komunikasi dan Informatika membantah adanya kebocoran data sedangkan sosok *hacker* dengan nama Bjorka memberikan dua juta sampel nomor HP dari lima operator seluler di Indonesia yang bisa diunduh bebas untuk membuktikan bahwa data itu asli (Maulida, 2022). Hal ini menyebabkan masyarakat mulai kehilangan kepercayaan pada pemerintahan yang seharusnya menjaga dan bertanggung jawab atas data warganya (CNN Indonesia, 2022). Dalam insiden ini pemerintahan juga harus mempertimbangkan opini publik dalam melakukan perbaikan untuk mendapatkan kepercayaan masyarakat kembali, terutama ketika sosok *hacker* Bjorka sedang meraih dukungan dari masyarakat walaupun tindakan membocorkan data pribadi berpotensi memicu banyak korban kejahatan seperti penipuan (CNN Indonesia, 2022). Kemunculan sosok *hacker* Bjorka ini menuai berbagai tanggapan dari masyarakat, sehingga opini publik dapat menjadi bahan pertimbangan pemerintahan dalam menanggapi insiden ini dengan baik. Pada era *digital* saat ini penyebaran informasi berjalan dengan cepat sehingga adanya berbagai tanggapan berupa opini negatif maupun positif, agar tindakan pemerintahan terhadap insiden ini dapat lebih baik dari sebelumnya, pemerintahan perlu mempertimbangkan kondisi maupun opini publik saat ini sehingga dalam mengetahui opini publik perlu dilakukan pengkajian yang melibatkan pemrosesan teks karena bentuk data opini publik yang belum terstruktur.

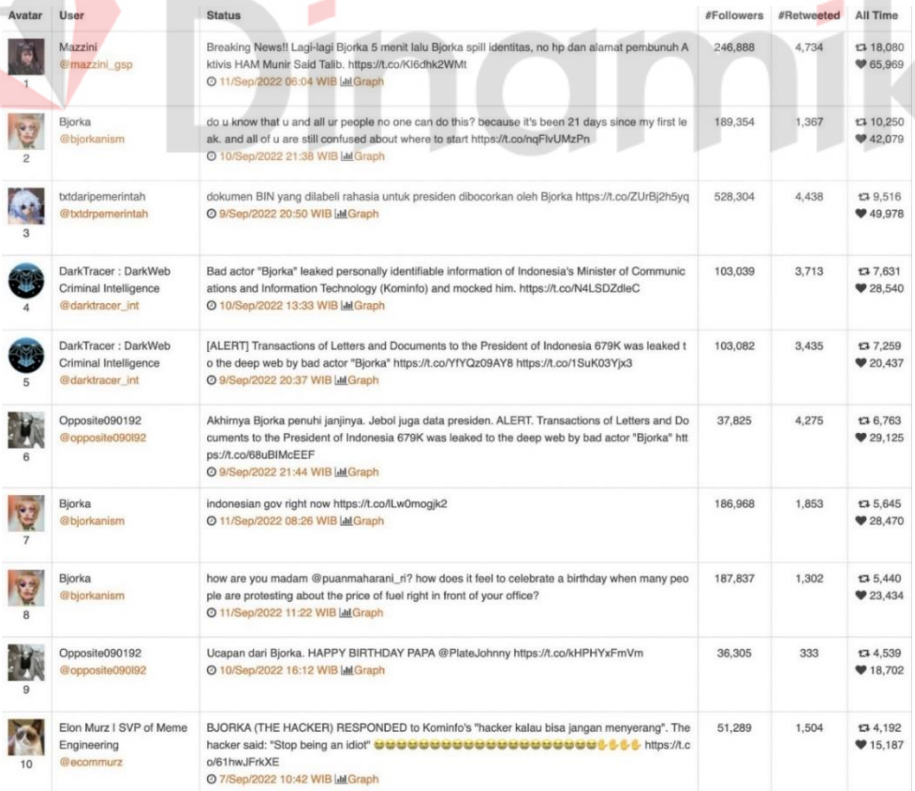
Analisis sentimen merupakan solusi untuk menyaring opini publik dan mengklasifikasikannya ke dalam kelas positif dan negatif, sehingga dari hasil klasifikasi tersebut dapat membantu pemerintahan untuk memberikan tanggapan yang lebih strategis dan tepat sehingga tidak dapat terjadinya keraguan oleh masyarakat terhadap instansi pemerintahan dalam menjaga data pribadi masyarakat maupun dapat menjadi salah satu rencana strategis ke depannya dalam menangani insiden yang sama. Analisis sentimen penelitian ini akan dilakukan pada periode bulan ketika insiden ini sedang ramai dibahas di *social media* yaitu bulan September tahun 2022. Menurut Fahmi (2022) kehadiran Bjorka sebagai

pelaku kebocoran data KOMINFO sering dibahas dibeberapa media sosial dalam minggu pertama September menurut data statistik berikut.



Gambar 1.1 Tren topik yang di-mention dibeberapa media sosial
(Sumber: Fahmi, 2022)

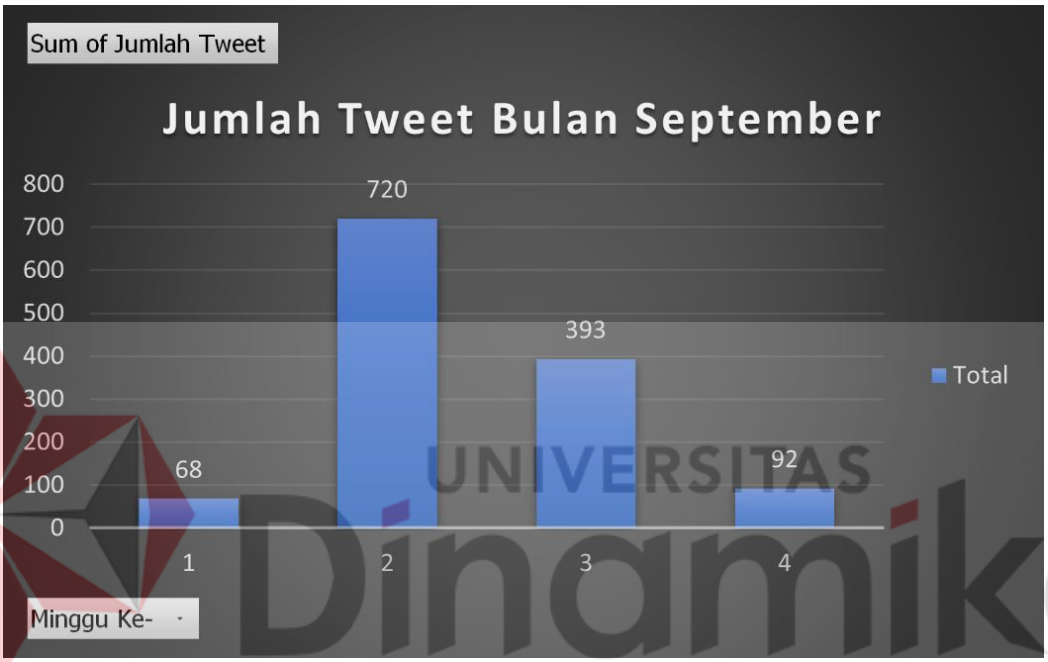
Salah satu media sosial yang digunakan yang paling banyak membahas insiden ini merupakan Twitter, sehingga terdapat beberapa *tweet* atau cuitan dari media sosial tersebut yang populer maupun cukup ramai dibahas seperti yang terlihat di Gambar 1.2.



Avatar	User	Status	#Followers	#Retweeted	All Time
1	Mazzini @mazzini_gsp	Breaking News!! Lagi-lagi Bjorka 5 menit lalu Bjorka spill identitas, no hp dan alamat pembunuh Aktivis HAM Munir Said Talib. https://t.co/Kl6dhk2WMt 11/Sep/2022 06:04 WIB Graph	246,888	4,734	18,080 65,969
2	Bjorka @bjorkanism	do u know that u and all ur people no one can do this? because it's been 21 days since my first leak. and all of u are still confused about where to start https://t.co/nqFlvUMzPn 10/Sep/2022 21:38 WIB Graph	189,354	1,367	10,250 42,079
3	txtdaripemerintah @txtdrpemerintah	dokumen BIN yang dilabeli rahasia untuk presiden dibocorkan oleh Bjorka https://t.co/ZUrBj2h5yq 9/Sep/2022 20:50 WIB Graph	528,304	4,438	9,516 49,978
4	DarkTracer : DarkWeb Criminal Intelligence @darktracer_int	Bad actor "Bjorka" leaked personally identifiable information of Indonesia's Minister of Communications and Information Technology (Kominfo) and mocked him. https://t.co/N4LSDZdleC 10/Sep/2022 13:33 WIB Graph	103,039	3,713	7,631 28,540
5	DarkTracer : DarkWeb Criminal Intelligence @darktracer_int	[ALERT] Transactions of Letters and Documents to the President of Indonesia 679K was leaked to the deep web by bad actor "Bjorka" https://t.co/YfYQz09AY8 https://t.co/1SuK03Yjx3 9/Sep/2022 20:37 WIB Graph	103,082	3,435	7,259 20,437
6	Opposite090192 @opposite090l92	Akhirnya Bjorka penuhi janjinya. Jebol juga data presiden. ALERT. Transactions of Letters and Documents to the President of Indonesia 679K was leaked to the deep web by bad actor "Bjorka" https://t.co/68uBIMcEEF 9/Sep/2022 21:44 WIB Graph	37,825	4,275	6,763 29,125
7	Bjorka @bjorkanism	indonesian gov right now https://t.co/lLw0mogjk2 11/Sep/2022 08:26 WIB Graph	186,968	1,853	5,645 28,470
8	Bjorka @bjorkanism	how are you madam @puanmaharani_ri? how does it feel to celebrate a birthday when many people are protesting about the price of fuel right in front of your office? 11/Sep/2022 11:22 WIB Graph	187,837	1,302	5,440 23,434
9	Opposite090192 @opposite090l92	Ucapan dari Bjorka. HAPPY BIRTHDAY PAPA @PlateJohnny https://t.co/kHPHYxFmVm 10/Sep/2022 16:12 WIB Graph	36,305	333	4,539 18,702
10	Elon Murz I SVP of Meme Engineering @ecommurz	BJORKA (THE HACKER) RESPONDED to Kominfo's "hacker kalau bisa jangan menyerang". The hacker said: "Stop being an idiot" 😂😂😂😂😂😂😂😂😂😂😂😂😂😂😂😂😂😂😂😂 https://t.co/61hwJFrkXE 7/Sep/2022 10:42 WIB Graph	51,289	1,504	4,192 15,187

Gambar 1.2 Cuitan Populer
(Sumber: Fahmi, 2022)

Pada penelitian ini juga ditemukan bahwa dari hasil *crawling data* cuitan di Twitter selama periode yang sudah ditentukan dengan cuitan terbanyak pada minggu ke-2 di bulan September ketika insiden tersebut sedang ramai dibahas sesuai dengan data statistik yang sudah dibahas di atas. Berikut ini merupakan gambar grafik jumlah *tweet* pada bulan september dengan paling banyak pada minggu kedua september dengan sebanyak 720 *tweet* dari total 1273 *tweet* pada bulan september.



Gambar 1.3 Jumlah Tweet Bulan September

Dari data-data tersebut dapat ditarik kesimpulan bahwa insiden ini cukup menarik pembicaraan publik di media sosial terutama Twitter, sehingga dapat terjadinya berbagai macam tanggapan maupun opini publik yang bersifat positif maupun negatif terhadap kehadiran tokoh Bjorka tersebut.

Algoritma yang digunakan untuk menunjang penelitian ini adalah algoritma *Support Vector Machine* (SVM) karena untuk saat ini SVM merupakan *classifier* dengan performa terbaik berdasarkan penelitian (Wibowo dkk, 2021) dengan topik Perbandingan Algoritma Klasifikasi Sentimen Twitter terhadap Insiden Kebocoran Data Tokopedia, pada penelitian tersebut juga disebutkan bahwa *F-measure* atau *f1-score* (nilai yang menunjukkan rata-rata harmonik presisi dan *recall* yang dihitung) yang dimiliki SVM lebih tinggi dari kedua algoritma lainnya. Algoritma ini dapat digunakan untuk melakukan klasifikasi tanggapan masyarakat

Indonesia dengan Bjorka ke dalam sentimen positif dan negatif dengan sumber data dari media sosial Twitter. Data yang diambil akan dipisah menjadi *data training* dan *data testing* untuk kebutuhan proses pada algoritma *Support Vector Machine*. *Data Testing* akan digunakan oleh algoritma SVM untuk proses klasifikasi (hasil klasifikasi SVM), sedangkan *data training* akan digunakan sebagai parameter dalam melakukan klasifikasi (Taufik, 2018).

Penelitian ini menggunakan data dari Twitter yang digunakan untuk memahami tanggapan publik atas insiden kebocoran data yang dilakukan oleh Bjorka, apakah lebih banyak yang beranggapan positif maupun negatif sehingga dari hasil penelitian tersebut dapat menjadi bahan evaluasi maupun rencana strategis pemerintah dalam menangani insiden kebocoran data ke depannya. Berdasarkan penjelasan di atas, penelitian ini berupa analisis sentimen untuk mengolah tanggapan masyarakat Indonesia dengan melakukan klasifikasi yang menggunakan algoritma *Support Vector Machine* (SVM).

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, maka permasalahannya adalah bagaimana melakukan analisis sentimen publik berdasarkan opini publik terhadap Bjorka dalam kebocoran data KOMINFO menggunakan algoritma *Support Vector Machine*.

1.3 Batasan Masalah

Berdasarkan perumusan masalah di atas, terdapat batasan masalah pada penelitian ini adalah sebagai berikut:

1. Data sekunder yang digunakan dalam penelitian ini berasal dari media sosial Twitter.
2. Klasifikasi data menggunakan algoritma *Support Vector Machine*.
3. Dalam melakukan ekstraksi data, hanya menggunakan metode *text preprocessing* pada *text mining*.
4. Data yang digunakan untuk analisis sentimen berasal dari *tweet* pada Twitter dengan *keyword* “Bjorka kebocoran data”.

5. *Output* dari penelitian ini berupa hasil klasifikasi positif dan negatif, persentase opini positif dan negatif dalam bentuk *pie chart*, dan *wordcloud* yang menampilkan kata yang sering muncul pada data komentar positif maupun negatif.

1.4 Tujuan

Berdasarkan latar belakang dan rumusan masalah di atas maka tujuan dari penelitian ini adalah untuk menghasilkan analisis sentimen terkait insiden kebocoran data yang dilakukan oleh Bjorka dengan cara melakukan klasifikasi menggunakan *Support Vector Machine* (SVM) sehingga dapat ditemukan tanggapan masyarakat lebih ke arah negatif atau positif.

1.5 Manfaat

Berdasarkan analisis sentimen yang dilakukan, hasil analisis sentimen tersebut dapat digunakan sebagai bahan evaluasi atau rencana strategis pemerintah dalam menangani insiden kebocoran data ke depannya maupun edukasi sosial terhadap keamanan data dan kebocoran data.

BAB II
LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian terdahulu digunakan sebagai dasar acuan dalam menambah wawasan penulis.

Tabel 2.1 Penelitian Terdahulu

No.	Judul	Metode Penelitian	Hasil	Akurasi
1	Perbandingan Algoritma Klasifikasi Sentimen Twitter terhadap Insiden Kebocoran Data Tokopedia	*Random Forest, Support-Vector Machine, dan Logistic Regression*	*Support Vector machine* merupakan hasil *classifier* dengan performa terbaik karena dari total 494 *tweet* yang dianalisa, *classifier* ini memberikan *f1-score* tertinggi sebesar 0.503583	*Precision* = 0.559671; *Recall* = 0.489899; *f1* = 0.503583
	Oleh: N. Wibowo, T. Maulana, H. Muhammad dkk			
	Perbedaan: Penelitian tersebut bertujuan untuk mengetahui algoritma mana yang lebih baik untuk digunakan dalam analisis sentimen.			
2	Analisis Sentimen Publik terhadap Pelayanan Tes SWAB-PCR COVID-19 di Indonesia menggunakan Algoritma *Support Vector Machine*	*Support-Vector Machine*	Dari total 103 data *tweet* didapatkan persentase sentimen positif sebesar 54.4% dengan sejumlah 56 *tweet* positif dan persentase sentimen negatif sebesar 45.6% dengan sejumlah 44 *tweet* negatif yang divisualisasikan melalui *pie chart* berdasarkan data set dengan *keyword* pelayanan *swab* pcr.	*Accuracy* = 76%; *Precision* = 75%; *Recall* = 81%
	Oleh: A. Mukminin			
	Perbedaan: Penelitian tersebut melakukan analisis sentimen publik terhadap Pelayanan Tes SWAB-PCR COVID-19.			
3	Analisis Sentimen Publik terhadap Tokoh Publik Menggunakan Algoritma *Support Vector Machine* (SVM)	*Support Vector Machine*	Penulis berhasil mengimplementasikan algoritma SVM. Hal ini diperlihatkan dengan hasil akurasi menggunakan data *tweet* sebanyak 630 data, dimana 420 data adalah data *training* dan 210 adalah data *testing*.	Memiliki tingkat akurasi paling baik sekitar 81% dengan Kernel Sigmoid
	Oleh: I. Taufik			
	Perbedaan: Penelitian tersebut melakukan analisis sentimen publik terhadap Tokoh Publik dan tidak melakukan visualisasi data.			

2.2 Analisis Sentimen

Sebuah studi komputasi untuk mengenali dan mengekspresikan opini, sentimen, evaluasi, sikap, emosi, subjektifitas, penilaian atau pandangan yang terdapat dalam suatu teks biasa disebut *Sentiment analysis* (analisis sentimen) atau *opinion mining* (penambangan opini) (Latuny,2021).

Analisis sentimen digunakan dalam memahami dan mengolah data yang berbentuk teks untuk mendapatkan informasi dalam sebuah kalimat opini, sehingga tujuan dari analisis sentimen tersebut adalah untuk melihat kecenderungan opini (positif, negatif atau netral) atau pendapat terhadap suatu masalah atau objek oleh seseorang (Zalyhaty, 2021).

2.3 Twitter

Menurut Achsanty (2021), Twitter merupakan situs *microblog* yang memberikan fasilitas bagi pengguna untuk mengirimkan sebuah teks dengan panjang maksimal 140 karakter. Inti dari twitter adalah *tweet* atau dalam bahasa indonesia yaitu cuitan. Pada awalnya twitter dimaksudkan sebagai fasilitas untuk menjawab pertanyaan "*What are you doing?*", walaupun sebagian orang menggunakannya sebagai tempat bercerita maupun berbagi informasi yang ingin mereka bagi ke platform tersebut.

Sumber informasi yang dapat digunakan untuk melakukan penelitian ini dapat berasal dari informasi yang disebar luaskan oleh setiap pengguna di twitter dengan suatu API (*Application Program Interface*) yaitu REST API. REST API ini menggunakan *pull strategy*, sehingga untuk mengambil suatu data dari twitter pengguna harus secara eksplisit memintanya (*request*), hal ini menyebabkan akses dari setiap *tweet* membutuhkan hak akses berupa *access token, access token secret, consumer key,* dan *consumer secret* (Mesak dkk, 2017).

2.4 *Text Mining*

Sebuah proses dalam menemukan relasi, fakta maupun informasi yang sifatnya tersembunyi di sebuah teks saat pemrosesan dan analisis data dengan jumlah yang cukup besar, struktur teks yang kompleks, dimensi yang tinggi maupun data yang bersifat *noise* merupakan definisi dari *Text Mining* (Pravina dkk, 2019).

Data Mining, NLP (Natural Language Processing) dan lain-lainnya biasanya digabungkan ke dalam penyelesaian masalah Text Mining, sehingga terdapat tahapan-tahapan seperti text pre-processing, pembobotan teks ataupun ekstraksi teks menggunakan teknik tertentu.

2.5 *Text Pre-Processing*

Untuk membersihkan suatu data sebelum dilakukan pemrosesan teks lainnya, maka akan dilakukan sebuah proses yang dinamakan *Text Pre-Processing* dengan mengolah data yang berfokus pada pembersihan dan merapikan data yang bersifat noise atau informasi yang hilang atau tidak lengkap (Pravina dkk, 2019). Terdapat 5 tahap dalam proses t*ext pre-processing*, yaitu:

1. *Case Folding*

Tahapan ini bertujuan untuk mengubah huruf kapital atau huruf besar menjadi huruf kecil (menyamakan semua bentuk kata seperti menjadi huruf kecil semua) (Indraloka & Santosa, 2017).

2. *Cleansing*

Proses penghapusan karakter-karakter di luar huruf alfabet a-z (tanda baca juga) atau penghapusan karakter-karater selain huruf disebut sebagai proses *Cleansing* (Pravina dkk, 2019).

3. *Translation*

Analisis sentimen pada zaman sekarang sangatlah dinamis terutama pada bidang linguistik komputasi, terdapat berbagai bahasa yang digunakan di sosial media Twitter sehingga dibutuhkan *Translation* untuk menerjemahkan kalimat berbahasa Inggris ke Bahasa Indonesia. (Pravina dkk, 2019).

4. *Tokenizing*

Tahapan ini melakukan pemisahan setiap kata dalam suatu kalimat yang dipisahkan dengan tanda koma (,). Pemecahan kalimat dan kata (Indraloka & Santosa, 2017).

5. *Stopword Removal*

Daftar kata umum yang tidak memiliki arti penting merupakan *Stopword Removal*, sehingga kata umum akan dihapus yang berfungsi untuk mengurangi jumlah kata yang disimpan oleh *corpus* atau sistem (Mukminin, 2021).

6. *Stemming*

Stemming merupakan proses untuk merubah kata-kata menjadi kata dasarnya (stem word). Proses stemming pada Bahasa Indonesia cukup kompleks, karena harus melakukan penghilangan seluruh imbuhan pada kata-kata yang terpada setiap tweet (Pravina dkk, 2019).

2.6 Pembobotan *Term Frequency-Inverse Document Frequency*

Menurut Pravina dkk (2019) pembobotan *Term Frequency-Inverse Document Frequency* (TF-IDF) merupakan salah satu proses teknik ekstraksi fitur dengan suatu proses tersebut memberikan nilai pada masing-masing kata yang terdapat pada tweet yang dilatih (*training*), sehingga untuk mengetahui seberapa penting sebuah kata mewakili sebuah kalimat, akan dilakukan pembobotan atau perhitungan. Pemberian skor dalam TF-IDF berdasarkan frekuensi munculnya kata dalam suatu dokumen.

Rumus pembobotan TF-IDF dapat dilihat di bawah ini:

1. Menghitung term frequency (tft,d)

2. Menghitung weighting term frequency ($Wtft,d$)

$$Wtf_{t,d} = \begin{cases} 1 + \log_{10} tf_{t,d}, if\ tf_{t,d} > 0 \\ 0, otherwise \end{cases} \quad (1)$$

3. Menghitung document frequency (df)

4. Menghitung bobot inverse document frequency (idf)

$$idf_t = \log_{10}\left(\frac{N}{df_t}\right) \quad (2)$$

5. Menghitung nilai bobot TF-IDF

$$W_{t,d} = W_{tf_{t,d}} \times idf_t \quad (3)$$

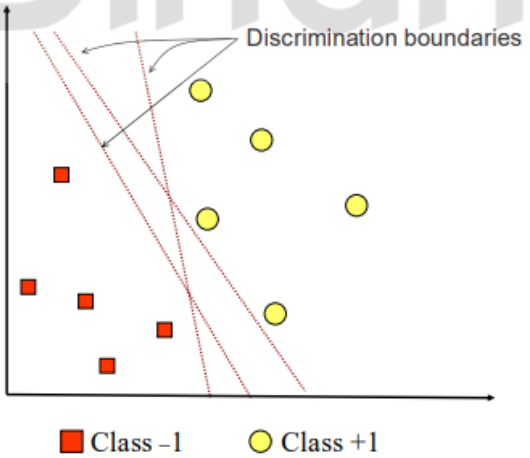
Keterangan:

$Wtf_{t,d}$	=	bobot kata dalam setiap dokumen
$tf_{t,d}$	=	jumlah kemunculan kata t dalam dokumen d
N	=	jumlah seluruh dokumen
df	=	jumlah dokumen yang mengandung term
idf_t	=	bobot inverse dari nilai df
$W_{t,d}$	=	bobot TF-IDF

2.7 *Support Vector Machine* (SVM)

Support Vector Machine ialah salah satu *machine learning* yang bisa memprediksi kelas menurut hasil proses *training*. Dengan melakukan *training* memakai data masukan dalam wujud numerik serta hasil dari ekstraksi fitur ataupun TF- IDF sehingga didapatkan suatu pola yang hendak digunakan buat pelabelan. Algoritma SVM berkaitan dengan *text mining*, sehingga kala informasi dari *text mining* telah ditemui hingga algoritma SVM hendak melaksanakan pengklasifikasian data dari hasil *text mining* tersebut(Pravina dkk, 2019).

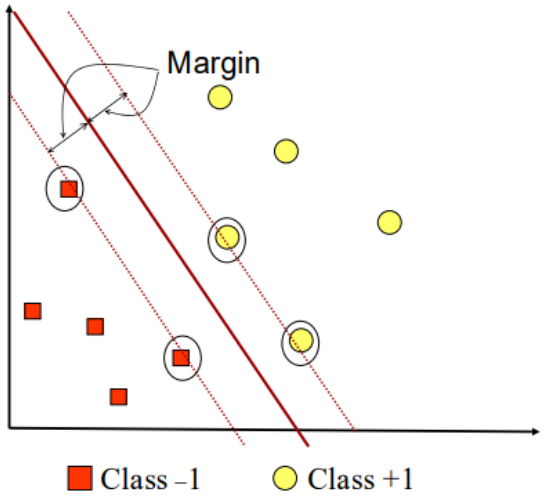
Menurut Pravina dkk (2019) konsep SVM dapat dijelaskan secara sederhana sebagai usaha mencari *hyperplane* (sebutan pemisah antar *class*) terbaik yang berfungsi sebagai pemisah dua buah *class* pada input *space*. Gambar 2.1 memperlihatkan beberapa pola yang merupakan anggota dari dua *class* : +1 dan -1. Pola yang tergabung dengan *class* -1 disimbolkan dengan warna merah kotak, sedangkan pola pada *class* +1, disimbolkan dengan warna kuning lingkaran. Problem klasifikasinya dapat diterjemahkan dengan usaha menemukan garis (*hyperplane*) yang memisahkan antara kedua kelompok tersebut. Berbagai alternatif garis pemisah (*discrimination boundaries*) juga ditunjukkan pada Gambar 2.1 di bawah ini.



Gambar 2.1 Proses SVM menemukan hyperplane terbaik

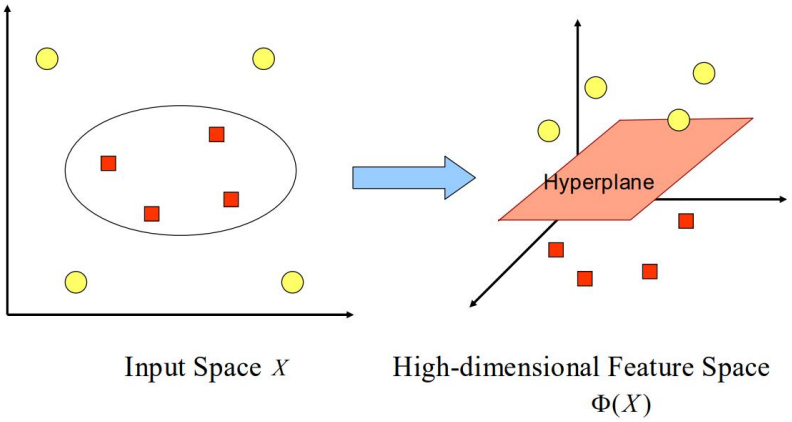
Hyperplane pemisah terbaik dapat ditemukan dengan mengukur *margin* dari *hyperplane* itu sendiri dan mencari titik maksimalnya. Margin disini merupakan jarak antara *hyperplane* tersebut dengan pola terdekat dari masing-masing *class*. Pola yang paling dekat itu disebut dengan *support vector* seperti yang ditunjukkan pada Gambar 2.2. Garis *hyperplane* yang terbaik dapat ditunjukkan terletak tepat

pada tengah-tengah antara kedua *class*. Usaha untuk mencari lokasi *hyperplane* merupakan inti dari proses pembelajaran SVM.



Gambar 2.2 *Hyperplane* terbaik diantara dua *class*

Prinsip dasar SVM adalah *linear classifier*, dan selanjutnya dikembangkan agar dapat bekerja pada problem *non-linear* dengan memasukkan konsep *kernel trick* pada ruang kerja berdimensi tinggi. Sebuah SVM dapat dimodifikasi dengan memasukkan fungsi Kernel. Hal ini digunakan karena kebanyakan permasalahan di dunia bersifat non-linear daripada linear, sehingga SVM membutuhkan untuk dimodifikasi dengan fungsi Kernel. Pada Gambar 2.3 menjelaskan bahwa dalam klasifikasi *non-linear*, pertama-tama data x dipetakan oleh fungsi Φ(x) ke ruang vektor yang berdimensi lebih tinggi, sehingga pada ruang vektor yang baru ini *hyperplane* yang memisahkan kedua *class* tersebut dapat dikonstruksikan. Pada Gambar 2.3 sebelah kiri diperlihatkan data pada *class* kuning dan data pada *class* merah yang berada pada input *space* berdimensi dua tidak dapat dipisahkan secara *linear*, lalu pada Gambar 2.3 sebelah kanan menunjukkan dengan fungsi Φ memetakan tiap data pada input *space* tersebut ke ruang vektor yang baru dengan dimensi yang lebih tinggi sehingga kedua *class* dapat dipisahkan secara *linear* oleh sebuah *hyperplane*.



Gambar 2.3 Fungsi Φ memetakan data ke ruang vektor yang lebih tinggi

Berikut ini merupakan Tahapan algoritma SVM dan konsep Kernel Trick yang umum digunakan menurut Trivusi (2022):

1. Kernel Linear

Fungsi kernel linear digunakan untuk klasifikasi data linear. Kernel linear digunakan ketika data yang dianalisis sudah terpisah secara linear. Kernel linear cocok ketika terdapat banyak fitur dikarenakan pemetaan ke ruang dimensi yang lebih tinggi tidak benar-benar meningkatkan kinerja. Persamaan fungsi kernel linear dapat dilihat pada gambar di bawah ini.

$$\mathrm{K}\,(x, xi) = sum(x * xi) \qquad (4)$$

2. Kernel Polynomial

Fungsi kernel polynomial merupakan fungsi kernel yang digunakan ketika data tidak terpisah secara linear. Dalam machine learning, kernel polynomial adalah fungsi kernel yang cocok untuk digunakan dalam SVM dan kernelisasi lainnya, di mana kernel mewakili kesamaan vektor sampel pelatihan dalam ruang fitur. Kernel polynomial juga cocok untuk memecahkan masalah klasifikasi pada dataset pelatihan yang dinormalisasi. Persamaan untuk fungsi kernel polynomial terdapat pada gambar di bawah ini.

$$\mathrm{K}\,(x, xi) = 1 + sum(x * xi)\ \hat{}\,d \qquad (5)$$

3. Kernel RBF (Radial Basic Function)

Kernel RBF atau juga disebut kernel Gaussian adalah konsep kernel yang paling banyak digunakan untuk memecahkan masalah klasifikasi data yang tidak dapat dipisahkan secara linear. Kernel ini dikenal memiliki performa

yang baik dengan parameter tertentu, dan hasil dari pelatihan memiliki nilai error yang kecil dibandingkan dengan kernel lainnya. Rumus persamaan untuk fungsi kernel RBF terdapat pada gambar di bawah ini.

$$\mathrm{K}\,(x, xi) = \exp\,(-gamma * sum(x - xi\text{^}2)) \qquad (6)$$

Tahapan algoritma SVM:

1. Melakukan input hasil perhitungan TF-IDF

2. Meminimalkan nilai *margin*

$$\frac{1}{2}\|W\|^2 = \frac{1}{2}(w_1^2 + w_2^2) \qquad (7)$$

Dengan catatan $y_i(w_i . x_i + b) \geq 1$ dan $i = 1,2,3, \ldots, N$

Keterangan:

y = kelas

x_i = data

w_i = bobot

b = bias

3. Menghitungkan nilai bobot (w)

4. Menghitungkan nilai bias (b)

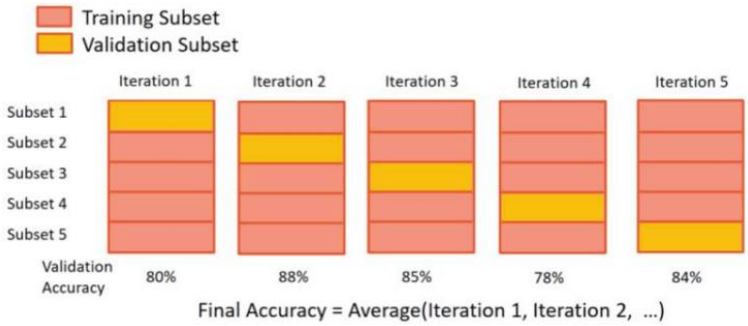
5. Mencari persamaan *hyperplane* dengan rumus

$$w_i + b = 0 \qquad (8)$$

6. Pengklasifikasian SVM

2.8 *K-Fold Cross Validation*

K-Fold Cross Validation merupakan pembagian data yang dilakukan dengan setiap subset (*fold*) yang sama kemudian akan dilihat bagaimana performa dari klasifikasi yang dilakukan sebelumnya, sehingga *K-Fold Cross Validation* menghasilkan nilai yang disebut *performance value*. Proses ini biasa disebut evaluasi performa, yaitu menguji hasil klasifikasi dengan mengukur nilai kinerja dari sistem yang dibuat (Anjasmoros dkk, 2020).



Gambar 2.4 Contoh Proses *K-Fold Cross Validation*

2.9 *Confusion Matrix*

Menurut Pravina dkk (2019) *Confusion Matrix* merupakan teknik yang digunakan untuk mengevaluasi klasifikasi model untuk memperkirakan mana objek yang benar dan mana objek yang salah. Matriks dari prediksi yang akan dibandingkan dengan kelas asli berisi berupa informasi aktual dan prediksi nilai klasifikasi, kemudian sistem berhasil melakukan pengklasifikasian *tweet* tersebut sehingga membutuhkan ukuran untuk menentukan seberapa valid atau tepat klasifikasi telah dibuat oleh sistem. Tabel 2.2 di bawah ini menunjukkan *confusion matrix* yang digunakan untuk membantu dalam perhitungan sistem evaluasi.

Tabel 2.2 *Confusion Matrix*

Classification	*Predicted Positives*	*Predicted Negatives*
Actual Positive Cases	*Number of True Positive Cases (TP)*	*Number of False Negative Cases (FN)*
Actual Negative Cases	*Number of False Positive Cases (FP)*	*Number of True Negative Cases (TN)*

Dalam pengujian akurasi ini menggunakan *confusion matrix* empat kondisi sebagai berikut: *True Positive (TP), True Negatif (TN), False Positive (FP),* dan *False Negative (FN)*. Berdasarkan *matrix confusion* tersebut dapat dihitung nilai *F1, accuracy, precision,* dan *recall*. Perhitungan detailnya dapat dilihat di bawah ini:

F1-score adalah *harmonic mean* dari *precision* dan *recall*. Nilai terbaik *F1* adalah 1.0 dan nilai terburuknya 0, sehingga jika nilai *F1* memiliki skor yang baik berarti model klasifikasi yang digunakan memliki *precision* dan *recall* yang baik. Bentuk perhitungannya di bawah ini:

$$F1 = \frac{2*(Precision*Recall)}{(Precision+Recall)} \quad (9)$$

Accuracy menggambarkan seberapa akurat model dalam mengklasifikasikan dengan benar seperti perhitungan di bawah ini:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (10)$$

Precision menggambarkan akurasi antara data *request* dengan hasil prediksi yang diberikan oleh model seperti perhitungan di bawah ini:

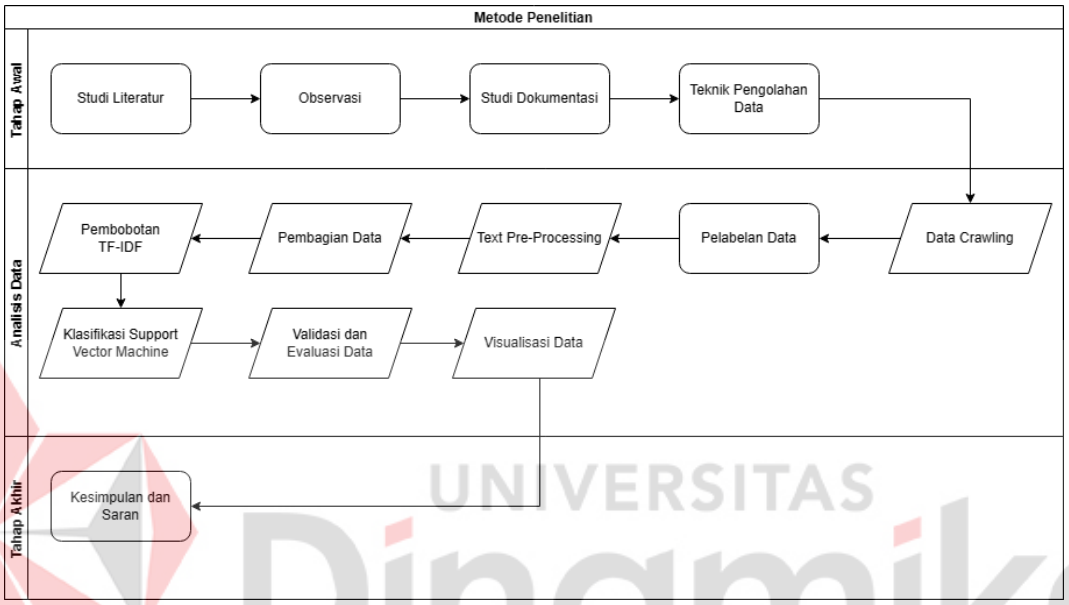
$$Precision = \frac{TP}{TP+FP} \quad (11)$$

Recall menggambarkan kesuksesan model dalam menemukan kembali sebuah informasi dengan perhitungan di bawah ini:

$$Recall = \frac{TP}{TP+FN} \quad (12)$$

BAB III
METODOLOGI PENELITIAN

Pada tahap ini, tahapan penelitian yang digunakan dibagi menjadi tiga bagian yaitu tahap awal, analisis data dan tahap akhir. Detailnya dapat dilihat pada gambar di bawah ini.

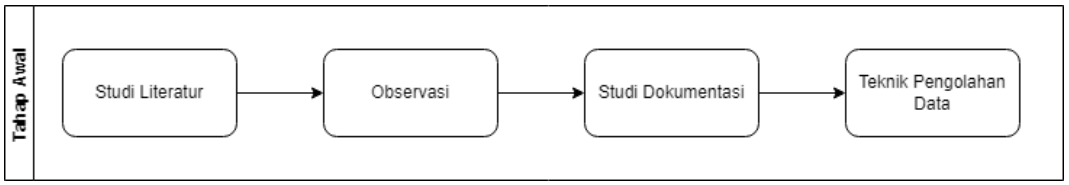


Gambar 3.1 Diagram Alir Metode Penelitian

Pada Gambar 3.1 terdapat proses yang dilakukan secara manual oleh peneliti atau sukarelawan dan proses yang dilakukan secara otomatis atau dilakukan oleh sistem. Proses yang dilakukan manual oleh manusia digambarkan dalam bentuk kotak, sedangkan untuk proses yang dilakukan oleh sistem digambarkan dalam bentuk jajar genjang.

3.1 Tahap Awal

Tahap awal merupakan tahapan pertama yang dilakukan dalam penelitian ini sebelum melakukan analisis data. Diagram alir pada tahap awal berikut ini dapat dilihat pada gambar berikut di bawah ini.



Gambar 3.2 Diagram Alir Tahap Awal

3.1.1 Studi Literatur

Studi literatur merupakan tahap pertama dalam penelitian ini untuk membantu dalam penulisan terhadap topik yang diambil. Pada tahap ini akan dilakukan analisis terhadap penelitian terdahulu maupun referensi jurnal lainnya yang dapat membantu dalam menyelesaikan penelitian ini.

3.1.2 Observasi

Observasi yang dilakukan pada tahap ini yaitu melakukan pengamatan, mempelajari, dan memahami objek atau topik yang diteliti. Dalam penelitian ini yaitu mengamati *tweet* pada twitter terhadap kata kunci seperti "Bjorka kebocoran data" yang dapat memuat beberapa cuitan dari berbagai pengguna twitter.

3.1.3 Studi Dokumentasi

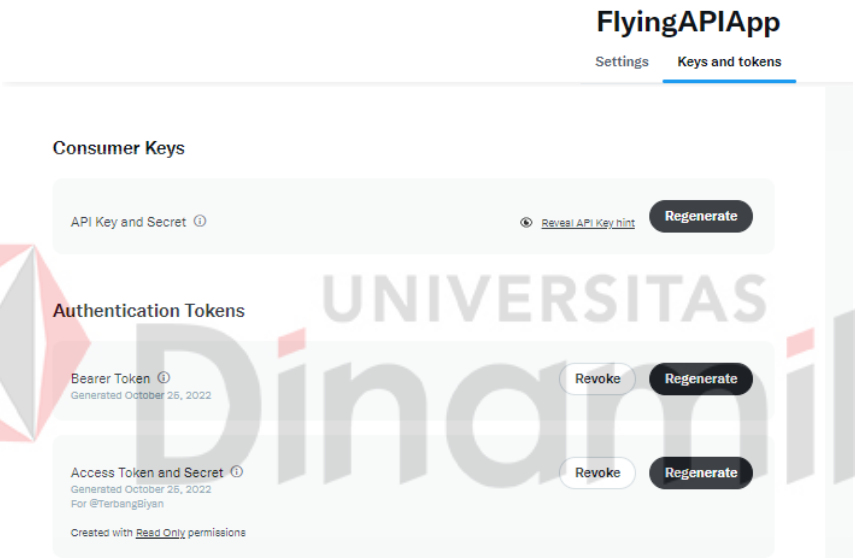
Studi dokumentasi merupakan pencarian opini atau *tweet* dari pengguna twitter terhadap kata kunci (*keyword*) seperti "Bjorka kebocoran data". Gambar di bawah ini merupakan contoh *tweet* berdasarkan *keyword* yang sudah ditentukan.

Gambar 3.3 Contoh *tweet* pro Bjorka

Gambar 3.4 Contoh *tweet* kontra Bjorka

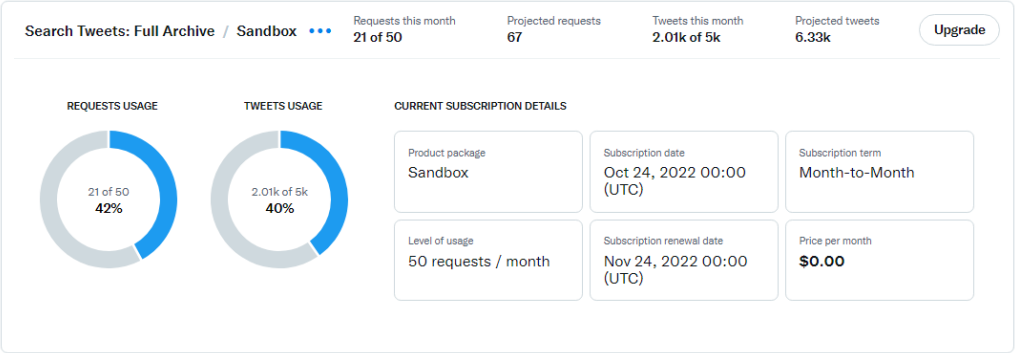
3.1.4 Teknik Pengolahan Data

Data yang digunakan guna pengolahan data ialah data cuitan pada Twitter yang sesuai dengan *keyword* yang telah ditetapkan. Langkah pertama yang dilakukan adalah meminta izin kepada Twitter dalam menggunakan *API*-nya sehingga diperlukan mengisi beberapa form maupun menunggu perizinan dari *Twitter for Developers*. Pada penelitian ini akan menggunakan Twitter API v2 versi *Elevated* karena peneliti hanya bisa melakukan akses pada versi tersebut saja, setelah mendapatkan aksesnya maka akan dilakukan penulisan kode untuk melakukan *crawling data* sesuai dengan aplikasi yang terdapat pada *Twitter for Developers* mulai dari *API Key and Secret*, *Bearer Token* dan *Access Token and Secret*.



Gambar 3.5 Contoh batasan *Search Tweets*

Akses Twitter pada *Elevated* cukup terbatas mulai dari hanya diperbolehkan melakukan *crawling data* sebanyak 5000 data dalam sebulan dengan akses *Full Archive* (akses ke *tweet* dalam jangka waktu sejak Twitter berdiri), berikut ini merupakan contoh batasan terhadap berapa banyak *tweet* yang bisa di-*crawling*.



Gambar 3.6 Contoh batasan *Search Tweets*

Seperti gambar di atas merupakan contoh nominal pencarian *tweet* dengan detail harga biaya langganannya, setelah melakukan pendaftaran *Twitter for Developers* maka dapat melanjutkan dengan melakukan koding menggunakan bahasa pemrograman *python* untuk melakukan proses *crawling data tweet* dari Twitter dan menghasilkan file tabel dengan format .xlsx atau .csv, kemudian dilanjutkan pada bagian Analisis Data.

3.1.5 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan dengan tujuan untuk mengetahui kebutuhan peneliti mulai dari *software*, *python* dan *library* yang digunakan beserta spesifikasi minimum pada *hardware* yang digunakan.

1. Kebutuhan Perangkat Lunak (*Software*)

   Kebutuhan perangkat lunak merupakan perangkat lunak yang digunakan dalam penelitian ini. Perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut:

   a. *PyCharm Community Edition* versi 2022.2.3 yang digunakan dalam proses *crawling*.
   b. *Google Colaboratory* yang digunakan untuk proses *text pre-processing* hingga visualisasi data.
   c. Bahasa pemrograman yang digunakan dalam penelitian ini merupakan *Python* versi 3.8.16 untuk *Google Colaboratory* dan *Python* versi 3.9 untuk *PyCharm*.
   d. *Library* tweepy versi 4.10.1, *library* ini digunakan saat *crawling data*.
   e. *Library* pandas versi 1.4.4, *library* ini digunakan untuk proses *import* maupun *export* data menjadi bentuk .csv.
   f. *Library* googletrans versi 3.1.0a0, *library* ini digunakan untuk proses *translate* pada *text pre-processing*.
   g. *Library* xlsxwriter versi 3.0.5, *library* ini digunakan untuk *export* data ke bentuk .xlsx.
   h. *Library* sastrawi versi 1.0.1, *library* ini digunakan untuk proses *stemming* pada *text pre-processing*.

i. *Library* nltk versi 3.7, *library* ini digunakan untuk proses *stopwords* dan *tokenizing* (pembobotan).

j. *Library* sklearn versi 0.0, *library* ini digunakan untuk proses klasifikasi SVM, *confusion matrix* dan *k-fold cross validation*.

k. *Library* matplotlib versi 3.6.0, *library* ini digunakan untuk proses visualisasi data berupa *pie chart*.

l. *Library* wordcloud versi 1.8.2.2, *library* ini digunakan untuk proses visualisasi data berupa *word cloud*.

2. Kebutuhan Perangkat Keras (*Hardware*)

Kebutuhan perangkat keras merupakan peralatan fisik yang mendukung proses penggunaan perangkat lunak dalam menjalankan fungsinya. Kebutuhan perangkat keras yang dibutuhkan adalah sebagai berikut:

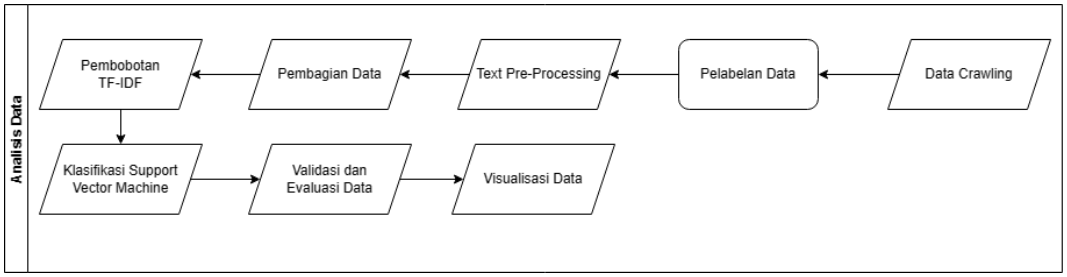
a. Komputer yang digunakan pada penelitian ini memiliki *processor* Intel® Core™ i5-8250U.

b. Komputer yang digunakan pada penelitian ini memiliki RAM (*Random Access Memory*) sebesar 8GB.

c. Komputer yang digunakan pada penelitian ini memiliki monitor dengan resolusi 1366 x 768 pixel.

d. Komputer yang digunakan pada penelitian ini memiliki GPU (*Graphics Processing Unit*) NVIDIA GeForce MX130.

3.2 Analisis Data



Gambar 3.7 Diagram Alir Analisis Data

Pada tahap analisis data bertujuan untuk mengetahui dan memahami kebutuhan yang diperlukan dalam melakukan analisis, kemudian dilakukan pemrosesan data.

3.2.1 *Data Crawling*

Pada tahap ini dilakukan penambangan atau pengambilan data menggunakan bahasa pemrograman *python* dan dengan bantuan *software code editor* PyCharm. *API Token* yang sudah di-*request* pada halaman *Twitter for Developers* dimasukkan ke *code* untuk mendapatkan akses koneksi terhadap data-data *tweet*, kemudian melakukan *import library* seperti tweepy dan pandas. Tweepy merupakan *library* dari *python* yang sudah disediakan oleh pihak Twitter untuk dapat mengakses dan mengambil data-data yang ada di dalam Twitter dengan menggunakan operator-operator khusus, sedangkan Pandas merupakan salah satu *library* dari *python* yang salah satu fungsinya merupakan memanipulasi data, *library* Pandas digunakan untuk mengubah data *crawling* Twitter menjadi format .csv, kemudian dilakukan pencarian dengan memasukkan *keyword* yang sudah ditentukan dan menambahkan operator *Remove Duplicate* untuk menghapus *tweet* yang bersifat kembar, kemudian hasilnya akan disimpan dalam bentuk *Comma-separated values* atau .csv yang akan di-*import* ke *Excel* untuk dijadikan bentuk tabel yang mudah dibaca pada *Excel*. Data hasil *crawling* yang diperoleh sebanyak 1300 data. Berikut ini contoh data yang hasil berhasil di-*crawling*.

3.2.2 Pelabelan Data

Pelabelan data merupakan penentuan kelas atribut yang dilakukan berdasarkan subjektifitas peneliti dengan bantuan 2 sukarelawan yang memiliki pemahaman cukup terhadap penelitian ini. Tahapan ini dilakukan dengan melakukan pelabelan pada masing-masing *tweet* dengan label negatif dan positif berdasarkan pencocokan kamus kata positif dan negatif dari forum *github List of Opinion Words (positive/negative) in Bahasa Indonesia for Sentiment Analysis* oleh Devid (2017). Contoh dari tahap pelabelan data terdapat pada tabel di bawah ini.

Tabel 3.1 Contoh Pelabelan Data

Tweet	Label
Ya kalo menurut bjorka itu dokumen rahasia menurut gw isinya masih yg kategori umum sih	Negatif
Pada akhirnya skema Bjorka berhasil, mudah2an kasus kebocoran data ini bisa jd evaluasi bagi pemerintah	Positif

Bjorka ini trending soal kebocoran data ya? Apa ngaruh buat rakyat juga?	Negatif

3.2.3 *Text Pre-Processing*

Tahap ini merupakan proses untuk ekstraksi data dengan tujuan untuk mengubah data dari format yang tidak terstruktur hingga menjadi format data yang tersusun dari kata dasar, proses ini dilakukan dengan *library* yang digunakan pada *Google Colab* menggunakan bahasa pemrograman *python*, proses tersebut terdiri dari *case folding* yang menggunakan *library re*, *cleansing* yang menggunakan *library re*, *translation* yang menggunakan *library GoogleTranslate*, *tokenizing* yang menggunakan *library nltk*, *stopword removal* yang menggunakan *library sastrawi*, *stemming* yang menggunakan *library sastrawi*.

3.2.4 Pembagian Data

Tahap ini merupakan proses untuk membagi data menjadi *data training* dan *data testing*. *Data training* merupakan data yang digunakan untuk melakukan proses *training* pada algoritma SVM sedangkan *data testing* akan digunakan untuk melakukan *testing* pada SVM setelah melakukan *training*. Proses ini dilakukan dengan *library* yang digunakan pada *Google Colab* menggunakan bahasa pemrograman *python*, *library* tersebut merupakan *Sklearn* dengan menggunakan *train_test_split*.

3.2.5 Pembobotan TF-IDF

Dari tahap pembagian data kemudian dilakukan pembobotan TF-IDF. Proses TF-IDF di *phyton* menggunakan *library TfidfVectorizer*, dengan tujuan mengubah data teks menjadi data numerik agar dapat dilakukan perhitungan bobot setiap kata, sehingga setiap semakin besar bobot sebuah kata maka kata tersebut dianggap penting. Pembobotan ini dapat digunakan sebagai penyaringan data agar ketika ada kata yang bernilai 0 maka tidak akan diproses maupun ditampilkan untuk tahap selanjutnya.

3.2.6 Klasifikasi *Support Vector Machine*

Dalam penerapan algoritma *Support Vector Machine (SVM)* memanfaatkan *library sklearn* yang merupakan *library* berbasis *python* yaitu *library scikit-learn*. Pada penelitian ini menggunakan kernel linear, karena penggunaan kernel linear lebih simpel dan dataset yang digunakan pada penelitian ini dapat dipisah dengan mudah secara linear (Raschka, 2016). Perhitungan TF-IDF pada proses sebelumnya akan digunakan untuk meminimalkan nilai *margin* kemudian menghitung nilai bobot dan nilai bias sehingga baru dapat dilakukan pencarian persamaan *hyperplane*, maka dari garis *hyperplane* yang ditemukan maka peneliti dapat menentukan masing-masing kategori kelas positif dan kelas negatif.

3.2.7 Validasi dan Evaluasi Data

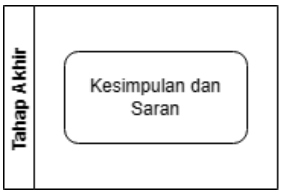
Hasil pengujian dari klasifikasi SVM akan dilakukan proses validasi dan evaluasi untuk mengetahui seberapa akurat hasil yang pengujian yang dilakukan. Dalam proses melakukan validasi menggunakan *K-Fold Cross Validation* yang menggunakan *library sklearn* dengan membagi data menjadi beberapa bagian, lalu untuk evaluasinya menggunakan *Confusion Matrix* untuk mengukur akurasi algoritma.

3.2.8 Visualisasi Data

Visualisasi data bertujuan untuk menampilkan data hasil akhir dengan bentuk *wordcloud* dan diagram *pie chart* untuk membantu pembaca dan peneliti dengan mudah memahami hasil. Perkata yang kerap timbul dalam kalimat ataupun teks yang dianalisis akan ditampikan dengan bentuk visualisasi WordCloud, sehingga semakin besar sebuah kata-kata pada WordCloud maka semakin sering suatu kata muncul dalam kalimat atau teks.. Visualisasi *pie chart* menampilkan persentase hasil klasifikasi kelas positif dan kelas negatif. Visualisasi *Word Cloud* menggunakan *library WordCloud* di *python*, sedangkan untuk visualisasi *Pie Chart* menggunakan *library matplotlib*.

3.3 Tahap Akhir

Pada tahap akhir penelitian ini berisikan kesimpulan dan saran untuk pembaca maupun peneliti agar penelitian selanjutnya maupun penelitian lainnya bisa lebih baik lagi.



Gambar 3.8 Diagram Alir Tahap Akhir

3.3.1 Kesimpulan

Pada tahap kesimpulan ini akan dijelaskan mengenai hasil analisis sentimen yang menggunakan algoritma SVM dengan mengetahui hasil klasifikasi data yang dibagi menjadi kelas positif maupun negatif berdasarkan data opini publik terhadap Bjorka dalam insiden kebocoran data KOMINFO melalui cuitan di twitter.

3.3.2 Saran

Pada tahap saran bertujuan untuk memberikan masukan kepada pembaca untuk melakukan penelitian yang lebih baik maupun menjadi acuan penelitian lain dengan topik yang serupa.

BAB IV
HASIL DAN PEMBAHASAN

Tahap ini akan membahas hasil dan pembahasan penelitian yang diimplementasikan berdasarkan tahapan-tahapan yang sudah ditentukan sesuai dengan metode yang digunakan.

4.1 *Data Crawling*

Pengambilan data dilakukan menggunakan bahasa pemrograman *python* dengan melakukan *request API Token* ke Twitter terlebih dahulu dan memasukkannya pada *config.py* pada *software code editor* PyCharm untuk proses autentikasi seperti pada Gambar 4.2, kemudian memasukkan kata kunci yang digunakan dalam pencarian serta memasukkan label pada *environment project* yang sudah dibuat sebelumnya pada tahap Teknik Pengolahan Data. Ditambahkan juga operator untuk mendapatkan *tweet* seutuhnya tanpa dipotong, yaitu operator *tweet.extended_tweet['full_text']* dan ditambahkan *drop_duplicates* untuk mengurangi *tweet* yang kembar. Penulisan kodenya dapat dilihat pada Gambar 4.1 di bawah ini.

```
config.py   search.py
import csv
import tweepy
import config
import pandas as pd

auth = tweepy.OAuthHandler(config.API_KEY, config.API_SECRET)
auth.set_access_token(config.ACCESS_TOKEN, config.ACCESS_TOKEN_SECRET)
client = tweepy.API(auth)

# label = '30Days'
label = 'FullArch'
query = 'bjorka kebocoran data'

start_time = '202209010000'
end_time = '202210250000'
response = tweepy.Cursor(client.search_full_archive, label=label, query=query, maxResults=100, fromDate=start_time, toDate=end_time).items(100000)
columns = ['Time', 'User', 'Tweet']
data = []

for tweet in response:
    if tweet.truncated:
        tweet.text = tweet.extended_tweet['full_text']
    else:
        tweet.text = tweet.text

    data.append([tweet.created_at, tweet.user.screen_name, tweet.text])
    df = pd.DataFrame(data, columns=columns)
    print(df)
    df.drop_duplicates(subset=['Tweet'], keep='first', inplace=True)
    df.to_csv('tweets.csv')
```

Gambar 4.1 Proses *crawling* data

```
config.py   search.py
API_KEY = 'ZBFaVpp423vXPgFQLonUa62lD'
API_SECRET = 'u6udOtOcTpng3VmjaVFmst3YoyE7oXSqGexwB07NHbje4sCBAu'
BEARER_TOKEN = 'AAAAAAAAAAAAAAAAAAAAABQ%2BigEAAAAAOuy6DCF9ehYkoGX6d1GbzRHEPFc%3DqMFUWXMxWDqUOOe741EBN0VSU8GJMxeVP54xUkHjv9tHpJ78rW'
ACCESS_TOKEN = '210370589-8BPirceDBi9xmpMQWoM6bHIGpWJVuWVcngOs3Wpq'
ACCESS_TOKEN_SECRET = 'EXS9HM38UlgW6EzNEqeNvwzTdXGdeTJXVp3NB6AUbbtzx'
```

Gambar 4.2 Kode autentikasi yang digunakan

Data yang berhasil diambil melalui twitter akan disimpan dalam bentuk *Comma-separated values* (.csv) yang kemudian akan diubah menjadi *excel* (.xlsx) agar mempermudah peneliti dan sukarelawan lainnya dalam melakukan pelabelan data. Untuk contoh data yang diambil dalam bentuk *excel* dapat dilihat pada Gambar 4.3 di bawah ini.

	Time	User	Tweet
2	9/1/2022 12:22	buddykuofficial	Waspada! Dugaan kebocoran data pribadi terjadi lagi di situs Breached forum. Pengguna dengan nama Bjorka menjual data dengan judul 'Indonesia SIM Card (Phone Nu
3	9/1/2022 16:32	Spy_Zone85	Berdasarkan pengamatan atas penggalan data yang disebarkan oleh akun Bjorka, dapat disimpulkan bahwa data tersebut tidak berasal dari Kementerian
4	9/1/2022 16:56	sonyakei_	Berdasarkan pengamatan atas penggalan data yang disebarkan oleh akun Bjorka, dapat disimpulkan bahwa data tersebut tidak berasal dari Kementerian
5	9/1/2022 16:59	detikinet	Kominfo berkelit soal tuduhan akun Bjorka yang menyebutkan bahwa dugaan kebocoran data pendaftaran kartu SIM prabayar seluler itu berasal dari mereka.
6	9/1/2022 17:11	Elbarak_	Berdasarkan pengamatan atas penggalan data yang disebarkan oleh akun Bjorka, dapat disimpulkan bahwa data tersebut tidak berasal dari Kementerian Kominfo.
7	9/1/2022 17:12	detikinet	Kominfo berkelit soal tuduhan akun Bjorka yang menyebutkan bahwa dugaan kebocoran data pendaftaran kartu SIM prabayar seluler itu berasal dari mereka.
8	9/1/2022 18:06	detikcom	Kominfo berkelit soal tuduhan akun Bjorka yang menyebutkan bahwa dugaan kebocoran data pendaftaran kartu SIM prabayar seluler itu berasal dari mereka.
9	9/1/2022 18:38	AbahDeon	2.dan dijual di sebuah forum online "Breached Forums". Dugaan kebocoran data tersebut terungkap dari unggahan seorang anggota forum Breached, Bjorka pada 31
10	9/1/2022 18:39	AbahDeon	5.nomor HP yang dibagikan Bjorka merupakan asli milik seseorang. KompasTekno sudah berusaha menghubungi pihak Kementerian Kominfo @kemenkominfo
11	9/1/2022 19:19	JAckBianto	Kominfo berkelit soal tuduhan akun Bjorka yang menyebutkan bahwa dugaan kebocoran data pendaftaran kartu SIM prabayar seluler itu berasal dari mereka.
12	9/1/2022 20:48	detikinet	Kominfo berkelit soal tuduhan akun Bjorka yang menyebutkan bahwa dugaan kebocoran data pendaftaran kartu SIM prabayar seluler itu berasal dari mereka.
13	9/1/2022 20:56	Iconman221	Lagi dan lagi berita kebocoran data muncul kembali, kali ini kebocoran data dari KOMINFOOOOOOO, terdengar kabar kalau kebocoran data ini meliputi NIK, Nomor
14	9/2/2022 8:06	vita_AVP	Keterangan Terkait Kebocoran Data Pendaftaran Kartu SIM Telepon Indonesia- Kominfo tidak memiliki aplikasi untuk menampung data- Penggalan data yang
15	9/2/2022 10:00	validnewsid	Sebaliknya, Kemenkominfo membantah bahwa data yang ditawarkan di forum kebocoran data (breach forum) oleh username Bjorka itu adalah data dari mereka.
16	9/2/2022 10:44	GarudaWisnu11	Dipastikan Data Sim Card Bocor Bukan dari Kominfo Kominfo tidak memiliki kaitan dengan akun Bjorka.
17	9/2/2022 13:16	miicies	Pernyataan terkait Dugaan Kebocoran Data Pendaftaran Kartu SIM Telepon Indonesia Berdasarkan pengamatan atas penggalan data yang disebarkan oleh akun Bjorka,
18	9/2/2022 16:39	catchmeupid	Data tersebut diperjualbelikan di forum 'http://breached.to' melalui seorang pengguna bernama Bjorka. Data yang disebut bocor berukuran 87 GB dengan berisi 1,3 mil

Gambar 4.3 Data *Tweet* hasil *Crawling*

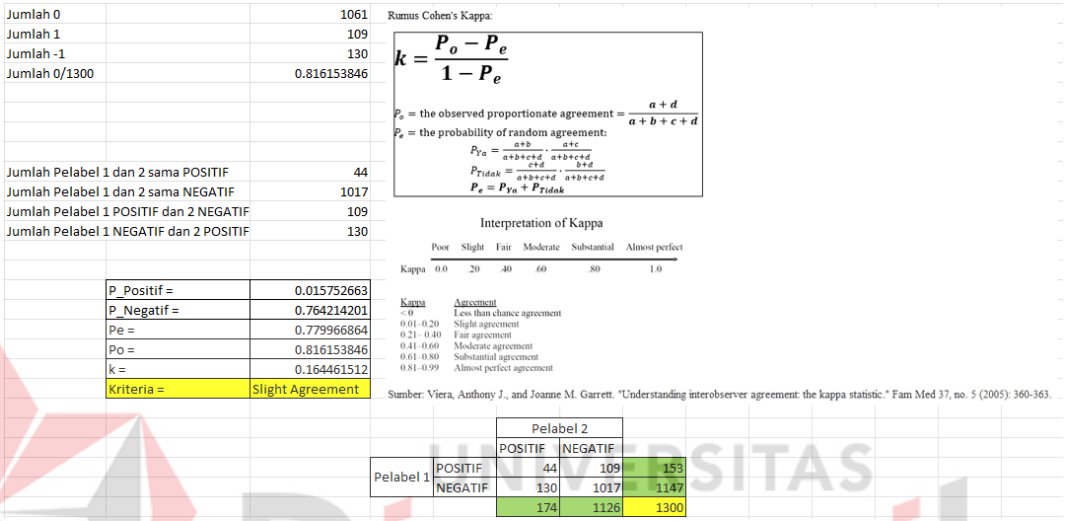
Data yang berhasil di-*crawling* sebanyak 1300 data dari periode 1 September 2022 hingga 25 Oktober 2022, dengan catatan banyak data yang sudah dihapus dengan operator *drop_duplicates*. 1300 data tersebut merupakan *crawling data* maksimal dari batas bulanan pengambilan data oleh Twitter pada *environments Elevated*, sehingga terdapat keterbatasan data dari penelitian ini.

Pada penelitian ini saat proses melakukan *crawling data* terjadi keterbatasan seperti data yang diambil hanyalah terbatas hingga 1300 data, hal ini disebabkan oleh melakukan *crawling data* secara terlambat oleh peneliti. Metode *crawling data* yang disarankan adalah melakukan *crawling data* saat data masih *fresh* pada hari itu juga, sehingga dapat melakukan *crawling data* lebih banyak dari penelitian ini untuk meningkatkan performa algoritma.

4.2 Pelabelan Data

Pelabelan data dilakukan dengan manual oleh peneliti dan 2 sukarelawan lainnya, dari 1300 data yang sudah diambil maka akan dilakukan pelabelan dengan setiap pelabel mendapatkan 1300 data. Data yang sudah dilabel oleh 2 sukarelawan akan diuji terlebih dahulu dengan uji reliabilitas menggunakan *Cohen's Kappa*

kemudian baru ditentukan label akhirnya. Dari hasil uji reliabilitas pada Gambar 4.4 menunjukkan nilai Kappa yang dihasilkan 0.16, hal ini menunjukkan bahwa dari kedua sukarelawan memiliki tingkat *Agreement* yang *Slight Agreement* atau kesepakatan yang kecil terhadap setiap pelabelan, sehingga dapat diartikan bahwa kedua sukarelawan kurang kompak dalam melakukan pelabelan. Dari pelabelan tersebut menghasilkan sebanyak 1164 *tweet* negatif dan 136 *tweet* positif. Gambar 4.5 di bawah ini merupakan beberapa contoh *tweet* hasil pelabelan.



Jumlah 0	1061
Jumlah 1	109
Jumlah -1	130
Jumlah 0/1300	0.816153846
Jumlah Pelabel 1 dan 2 sama POSITIF	44
Jumlah Pelabel 1 dan 2 sama NEGATIF	1017
Jumlah Pelabel 1 POSITIF dan 2 NEGATIF	109
Jumlah Pelabel 1 NEGATIF dan 2 POSITIF	130

P_Positif =	0.015752663
P_Negatif =	0.764214201
Pe =	0.779966864
Po =	0.816153846
k =	0.164461512
Kriteria =	Slight Agreement

Rumus Cohen's Kappa:

$$k = \frac{P_o - P_e}{1 - P_e}$$

P_o = the observed proportionate agreement = $\frac{a+d}{a+b+c+d}$

P_e = the probability of random agreement:

$$P_{Ya} = \frac{a+b}{a+b+c+d} \cdot \frac{a+c}{a+b+c+d}$$

$$P_{Tidak} = \frac{c+d}{a+b+c+d} \cdot \frac{b+d}{a+b+c+d}$$

$$P_e = P_{Ya} + P_{Tidak}$$

Interpretation of Kappa

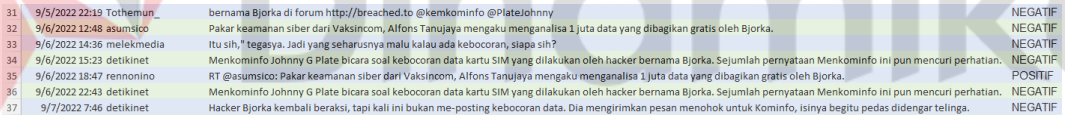
	Poor	Slight	Fair	Moderate	Substantial	Almost perfect
Kappa	0.0	.20	.40	.60	.80	1.0

Kappa	Agreement
< 0	Less than chance agreement
0.01–0.20	Slight agreement
0.21–0.40	Fair agreement
0.41–0.60	Moderate agreement
0.61–0.80	Substantial agreement
0.81–0.99	Almost perfect agreement

Sumber: Viera, Anthony J., and Joanne M. Garrett. "Understanding interobserver agreement: the kappa statistic." Fam Med 37, no. 5 (2005): 360-363.

		Pelabel 2		
		POSITIF	NEGATIF	
Pelabel 1	POSITIF	44	109	153
	NEGATIF	130	1017	1147
		174	1126	1300

Gambar 4.4 Hasil Uji Reliabilitas



31	9/5/2022 22:19	Tothemun_	bernama Bjorka di forum http://breached.to @kemkominfo @PlateJohnny	NEGATIF
32	9/6/2022 12:48	asumsico	Pakar keamanan siber dari Vaksincom, Alfons Tanujaya mengaku menganalisa 1 juta data yang dibagikan gratis oleh Bjorka.	NEGATIF
33	9/6/2022 14:36	melekmedia	Itu sih," tegasya. Jadi yang seharusnya malu kalau ada kebocoran, siapa sih?	NEGATIF
34	9/6/2022 15:23	detikinet	Menkominfo Johnny G Plate bicara soal kebocoran data kartu SIM yang dilakukan oleh hacker bernama Bjorka. Sejumlah pernyataan Menkominfo ini pun mencuri perhatian.	NEGATIF
35	9/6/2022 18:47	rennonino	RT @asumsico: Pakar keamanan siber dari Vaksincom, Alfons Tanujaya mengaku menganalisa 1 juta data yang dibagikan gratis oleh Bjorka.	POSITIF
36	9/6/2022 22:43	detikinet	Menkominfo Johnny G Plate bicara soal kebocoran data kartu SIM yang dilakukan oleh hacker bernama Bjorka. Sejumlah pernyataan Menkominfo ini pun mencuri perhatian.	NEGATIF
37	9/7/2022 7:46	detikinet	Hacker Bjorka kembali beraksi, tapi kali ini bukan me-posting kebocoran data. Dia mengirimkan pesan menohok untuk Kominfo, isinya begitu pedas didengar telinga.	NEGATIF

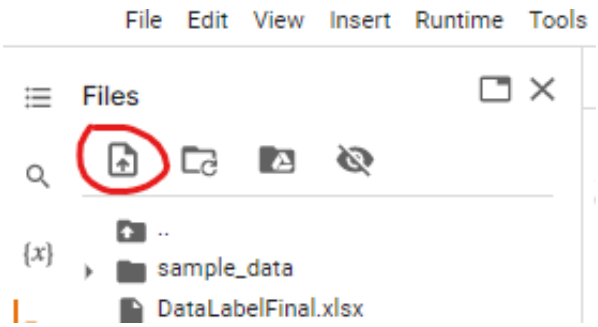
Gambar 4.5 Hasil Pelabelan

Proses pelabelan data berhasil dilakukan oleh peneliti bersama 2 sukarelawan lainnya, namun ditemukan permasalahan seperti terdapat kata-kata yang tidak tersedia pada kamus kata positif dan kamus kata negatif yang digunakan oleh peneliti dan 2 sukarelawan lainnya, sehingga jika tidak ada kata yang tergolong positif maupun negatif maka dimasukkan ke kategori positif karena tidak ada label netral pada penelitian agar pelabelan tidak lebih dari 2 kategori dan pelabel tidak bias terhadap label netral.

4.3 *Text Pre-Processing*

Tahap *Text Pre-Processing* yaitu untuk melakukan ekstraksi data. Proses ekstraksi data yang akan dilakukan dibagi menjadi 6 tahap secara berurutan. Pada tahap *Text Pre-Processing* menggunakan *code editor Google Colaboratory* dengan

bahasa *python*, sebelum melakukan *text pre-processing* maka akan dilakukan pengunggahan data *excel* yang sudah dilabel ke *Google Colaboratory* dengan tekan tombol pada lingkaran merah di Gambar 4.6 kemudian pilih *file* yang sudah ditentukan, dan akan muncul pada bagian bawah *sample_data* seperti Gambar 4.8 berikut.



Gambar 4.6 Contoh *upload* data *excel*

```
!pip install googletrans==3.1.0a0
!pip install xlsxwriter
!pip install Sastrawi
!pip install nltk
!pip install sklearn
!pip install matplotlib
!pip install wordcloud
```

Gambar 4.7 Instalasi *python library* yang akan digunakan

Setelah melakukan *upload* maka akan dilakukan pengunduhan *library python* yang akan digunakan seperti *googletrans*, *xlsxwriter*, *sastrawi*, *nltk*, *sklearn*, *matplotlib* dan *wordcloud* seperti pada Gambar 4.7. Setelah selesai mengunduh, berikutnya melakukan *import* beserta definisinya dan melakukan *download library python* yang akan digunakan dalam proses *Text Pre-Processing*, kemudian melakukan pemanggilan *file excel* yang sudah diunggah dan menghapus kolom yang tidak digunakan menggunakan operator *readexcel* dan *drop* seperti pada Gambar 4.9, untuk penulisan kode lebih lengkapnya pada Gambar L1.1. Berikut ini merupakan hasil data yang berhasil diunggah dengan kolom yang akan digunakan.

	Tweet	Label
0	Waspada! Dugaan kebocoran data pribadi terjadi...	NEGATIF
1	Berdasarkan pengamatan atas penggalan data yan...	POSITIF
2	Berdasarkan pengamatan atas penggalan data yan...	POSITIF
3	Kominfo berkelit soal tuduhan akun Bjorka yang...	NEGATIF
4	Berdasarkan pengamatan atas penggalan data yan...	NEGATIF
...	...	...
1295	RT @hiisyg96__: Indonesia masalahnya gak ada a...	NEGATIF
1296	MALING - RIBUT-RIBUT KEBOCORAN DATA OLEH BJORK...	NEGATIF
1297	Tau gak sih kalau ternyata Indonesia ada di po...	NEGATIF
1298	Pertanyaannya, seperti apa bentuk perlindungan...	NEGATIF
1299	@txtdrjkt Yg diblock mah akunnya bukan platfor...	NEGATIF

1300 rows × 2 columns

Gambar 4.8 Data yang berhasil diunggah

```
tweet = pd.read_excel("DataLabelFinal.xlsx")
tweet = tweet.drop(columns=['Time', 'User'])

tweet
```

Gambar 4.9 Proses *Upload* dan *drop* kolom

Setelah berhasil ditampilkan maka tahap selanjutnya yaitu mengubah label negatif dan positif menjadi bentuk angka, hasilnya dapat dilihat pada Gambar 4.10 di bawah ini. Untuk kode yang digunakan dapat dilihat pada Gambar 4.11.

	Tweet	Label
0	Waspada! Dugaan kebocoran data pribadi terjadi...	0
1	Berdasarkan pengamatan atas penggalan data yan...	1
2	Berdasarkan pengamatan atas penggalan data yan...	1
3	Kominfo berkelit soal tuduhan akun Bjorka yang...	0
4	Berdasarkan pengamatan atas penggalan data yan...	0
...	...	...
1295	RT @hiisyg96__: Indonesia masalahnya gak ada a...	0
1296	MALING - RIBUT-RIBUT KEBOCORAN DATA OLEH BJORK...	0
1297	Tau gak sih kalau ternyata Indonesia ada di po...	0
1298	Pertanyaannya, seperti apa bentuk perlindungan...	0
1299	@txtdrjkt Yg diblock mah akunnya bukan platfor...	0

1300 rows × 2 columns

Gambar 4.10 Hasil data perubahan kolom Label

```
Label = []
for index, row in tweet.iterrows():
  if row["Label"] == "POSITIF":
    Label.append(1)
  else:
    Label.append(0)

tweet["Label"] = Label
tweet
```

Gambar 4.11 Proses ubah kolom Label

Tahap berikutnya merupakan *Case Folding*, pada tahap ini dilakukan pengubahan setiap *tweet* yang memiliki huruf besar atau *uppercase* diubah menjadi huruf kecil atau *lowercase* menggunakan fungsi *.str.lower()* lalu data tersebut akan disimpan dalam bentuk *dataframe* menggunakan fungsi *pd.DataFrame* seperti Gambar 4.13. Berikut ini merupakan *DataFrame* yang sudah menggunakan huruf kecil semua.

	Tweet	Label
0	waspada! dugaan kebocoran data pribadi terjadi...	0
1	berdasarkan pengamatan atas penggalan data yan...	1
2	berdasarkan pengamatan atas penggalan data yan...	1
3	kominfo berkelit soal tuduhan akun bjorka yang...	0
4	berdasarkan pengamatan atas penggalan data yan...	0
...	...	...
1295	rt @hiisyg96__: indonesia masalahnya gak ada a...	0
1296	maling - ribut-ribut kebocoran data oleh bjork...	0
1297	tau gak sih kalau ternyata indonesia ada di po...	0
1298	pertanyaannya, seperti apa bentuk perlindungan...	0
1299	@txtdrjkt yg diblock mah akunnya bukan platfor...	0

1300 rows × 2 columns

Gambar 4.12 Hasil proses *Case Folding*

```
tweet['Tweet'] = tweet['Tweet'].str.lower()
df = pd.DataFrame(tweet[['Tweet','Label']])
df
```

Gambar 4.13 Proses *Case Folding*

Berikutnya akan dilakukan *Translation*, pada tahap ini akan dilakukan pengubahan *tweet* yang berbahasa Inggris menjadi bahasa Indonesia dengan bantuan *library GoogleTrans* yang sudah diunduh dengan kode pada Gambar 4.7, lalu untuk penulisan kode untuk pengubahan bahasa dapat dilihat pada Gambar

4.15. Hasil dari *tweet* yang sudah diubah bahasanya akan ditampilkan pada kolom baru agar dapat terlihat perbandingannya seperti Gambar 4.14 di bawah ini.

	Tweet	Label	translate
0	waspada! dugaan kebocoran data pribadi terjadi...	0	waspada! dugaan kebocoran data pribadi terjadi...
1	berdasarkan pengamatan atas penggalan data yan...	1	berdasarkan pengamatan atas penggalan data yan...
2	berdasarkan pengamatan atas penggalan data yan...	1	berdasarkan pengamatan atas penggalan data yan...
3	kominfo berkelit soal tuduhan akun bjorka yang...	0	kominfo berkelit soal tuduhan akun bjorka yang...
4	berdasarkan pengamatan atas penggalan data yan...	0	berdasarkan pengamatan atas penggalan data yan...
...	...	...	...
1295	rt @hiisyg96__: indonesia masalahnya gak ada a...	0	rt @hiisyg96__: indonesia masalahnya gak ada a...
1296	maling - ribut-ribut kebocoran data oleh bjork...	0	maling - ribut-ribut kebocoran data oleh bjork...
1297	tau gak sih kalau ternyata indonesia ada di po...	0	tau gak sih kalau ternyata indonesia ada di po...
1298	pertanyaannya, seperti apa bentuk perlindungan...	0	pertanyaannya, seperti apa bentuk perlindungan...
1299	@txtdrjkt yg diblock mah akunnya bukan platfor...	0	@txtdrjkt yg diblock mah akunnya bukan platfor...

1300 rows × 3 columns

Gambar 4.14 Hasil proses *Translation*

```
translator = Translator()

df['Tweet'] = df['Tweet'].astype(str)
df['translate'] = df['Tweet'].apply(translator.translate, src='auto', dest='id').apply(getattr, args=('text',))
df
```

Gambar 4.15 Proses *Translation*

Pada proses *translation*, *tweet* berhasil diterjemahkan tetapi ditemukan beberapa *tweet* yang masih terlewat untuk diterjemahkan seperti kata "*hacker*". Hal ini dapat disebabkan karena kata *hacker* tersebut ditambahkan imbuhan maupun faktor-faktor lainnya.

Pada tahap berikutnya yaitu *Cleansing*. Tahap ini merupakan penghapusan kata seperti *username* akun, *retweet* (RT), tanda baca, emoji, tautan dan angka pada *tweet text*. Tahap pertama dalam melakukan *cleansing* yaitu menghapus *username* pada masing-masing *tweet* dan menampilkannya pada satu kolom baru untuk perbandingan, sehingga hasilnya dapat dilihat pada Gambar 4.16 di bawah ini. Bentuk penulisan kodenya dapat dilihat pada Gambar L1.2 pada lampiran.

	Tweet	Label	translate	remove_user
0	waspada! dugaan kebocoran data pribadi terjadi...	0	waspada! dugaan kebocoran data pribadi terjadi...	waspada! dugaan kebocoran data pribadi terjadi...
1	berdasarkan pengamatan atas penggalan data yan...	1	berdasarkan pengamatan atas penggalan data yan...	berdasarkan pengamatan atas penggalan data yan...
2	berdasarkan pengamatan atas penggalan data yan...	1	berdasarkan pengamatan atas penggalan data yan...	berdasarkan pengamatan atas penggalan data yan...
3	kominfo berkelit soal tuduhan akun bjorka yang...	0	kominfo berkelit soal tuduhan akun bjorka yang...	kominfo berkelit soal tuduhan akun bjorka yang...
4	berdasarkan pengamatan atas penggalan data yan...	0	berdasarkan pengamatan atas penggalan data yan...	berdasarkan pengamatan atas penggalan data yan...
...	...	...	...	...
1295	rt @hiisyg96__: indonesia masalahnya gak ada a...	0	rt @hiisyg96__: indonesia masalahnya gak ada a...	rt : indonesia masalahnya gak ada abis-abisnya...
1296	maling - ribut-ribut kebocoran data oleh bjork...	0	maling - ribut-ribut kebocoran data oleh bjork...	maling - ribut-ribut kebocoran data oleh bjork...
1297	tau gak sih kalau ternyata indonesia ada di po...	0	tau gak sih kalau ternyata indonesia ada di po...	tau gak sih kalau ternyata indonesia ada di po...
1298	pertanyaannya, seperti apa bentuk perlindungan...	0	pertanyaannya, seperti apa bentuk perlindungan...	pertanyaannya, seperti apa bentuk perlindungan...
1299	@txtdrjkt yg diblock mah akunnya bukan platfor...	0	@txtdrjkt yg diblock mah akunnya bukan platfor...	yg diblock mah akunnya bukan platformnya. mbo...

1300 rows × 4 columns

Gambar 4.16 Hasil proses menghapus *username*

Kemudian melanjutkan proses *Cleansing* dengan menghapus emoji, tanda baca, kata *retweet* (rt) dan tautan, lalu dilanjutkan dengan proses *Tokenizing* yang kemudian akan dilanjutkan dengan *looping data array* untuk dilakukan proses *Stopword Removal* dan *Stemming*. Penulisan kode untuk proses tersebut dapat dilihat pada Gambar 4.18, sedangkan untuk hasilnya dapat dilihat pada Gambar 4.17.

	Tweet	Label	translate	remove_user	tweet_clean
0	waspada! dugaan kebocoran data pribadi terjadi...	0	waspada! dugaan kebocoran data pribadi terjadi...	waspada! dugaan kebocoran data pribadi terjadi...	[waspada, duga, bocor, data, pribadi, situs, b...
1	berdasarkan pengamatan atas penggalan data yan...	1	berdasarkan pengamatan atas penggalan data yan...	berdasarkan pengamatan atas penggalan data yan...	[dasar, amat, penggal, data, sebar, akun, bjor...
2	berdasarkan pengamatan atas penggalan data yan...	1	berdasarkan pengamatan atas penggalan data yan...	berdasarkan pengamatan atas penggalan data yan...	[dasar, amat, penggal, data, sebar, akun, bjor...
3	kominfo berkelit soal tuduhan akun bjorka yang...	0	kominfo berkelit soal tuduhan akun bjorka yang...	kominfo berkelit soal tuduhan akun bjorka yang...	[kominfo, kelit, tuduh, akun, bjorka, duga, bo...
4	berdasarkan pengamatan atas penggalan data yan...	0	berdasarkan pengamatan atas penggalan data yan...	berdasarkan pengamatan atas penggalan data yan...	[dasar, amat, penggal, data, sebar, akun, bjor...
...	...	...	...	...	...
1295	rt @hiisyg96__: indonesia masalahnya gak ada a...	0	rt @hiisyg96__: indonesia masalahnya gak ada a...	rt : indonesia masalahnya gak ada abis-abisnya...	[indonesia, gak, abis, abis, yah, brigadir, j,...
1296	maling - ribut-ribut kebocoran data oleh bjork...	0	maling - ribut-ribut kebocoran data oleh bjork...	maling - ribut-ribut kebocoran data oleh bjork...	[maling, ribut, ribut, bocor, data, bjorka, bo...
1297	tau gak sih kalau ternyata indonesia ada di po...	0	tau gak sih kalau ternyata indonesia ada di po...	tau gak sih kalau ternyata indonesia ada di po...	[tau, gak, sih, indonesia, posisi, daftar, neg...
1298	pertanyaannya, seperti apa bentuk perlindungan...	0	pertanyaannya, seperti apa bentuk perlindungan...	pertanyaannya, seperti apa bentuk perlindungan...	[tanya, bentuk, lindung, data, susah, payah, k...
1299	@txtdrjkt yg diblock mah akunnya bukan platfor...	0	@txtdrjkt yg diblock mah akunnya bukan platfor...	yg diblock mah akunnya bukan platformnya. mbo...	[yg, diblock, mah, akun, platform, mbok, ya, m...

1300 rows × 5 columns

Gambar 4.17 Hasil *Cleansing*, *Tokenizing*, *Stopword Removal* dan *Stemming*

Pada Gambar 4.17 dalam kolom *tweet_clean* ditemukan data yang berhasil dilakukan proses *tokenizing* masih memiliki tanda baca seperti koma (,) dan kurung siku ([]), sehingga di proses berikutnya dilakukan pembersihan tanda baca tersebut.

```python
# Happy Emoticons
emoticons_happy = set([
    ':-)', ':)', ';)', ':o)', ':]', ':3', ':c)', ':>', '=]', '8)', '=)', ':}',
    ':^)', ':-D', ':D', '8-D', '8D', 'x-D', 'xD', 'X-D', 'XD', '=-D', '=D',
    '=-3', '=3', ':-))', ":'-)", ":')", ':*', ':^*', '>:P', ':-P', ':P', 'X-P',
    'x-p', 'xp', 'XP', ':-p', ':p', '=p', ':-b', ':b', '>:)', '>;)', '>:-)',
    '<3'
    ])

# Sad Emoticons
emoticons_sad = set([
    ':L', ':-/', '>:/', ':S', '>:[', ':@', ':-(', ':[', ':-||', '=L', ':<',
    ':-[', ':-<', '=\\', '=/', '>:(', ':(', '>.<', ":'-(", ":'(", ':\\', ':-c',
    ':c', ':{', '>:\\', ';('
    ])

# all emoticons (happy + sad)
emoticons = emoticons_happy.union(emoticons_sad)

def clean_tweets(tweet):

    # remove old style retweet text "rt"
    tweet = re.sub(r'^rt[\s]+', '', tweet)

    # remove hyperlinks
    tweet = re.sub(r'https?:\/\/.*[\r\n]*', '', tweet)

    # remove hashtags
    tweet = re.sub(r'#', '', tweet)

    #remove coma
    tweet = re.sub(r',','',tweet)

    #remove angka
    tweet = re.sub('[0-9]+', '', tweet)

    #remove enter
    tweet = tweet.replace('\n', ' ')

    #remove tanda baca
    remove = string.punctuation
    translator = str.maketrans(remove, ' '*len(remove))
    tweet = tweet.translate(translator)

    # tokenize tweets
    tokenizer = TweetTokenizer(preserve_case=False, strip_handles=True, reduce_len=True)
    tweet_tokens = tokenizer.tokenize(tweet)

    tweets_clean = []
    for word in tweet_tokens:
        if (word not in stopwords_indonesia and # remove stopwords
                word not in emoticons and # remove emoticons
                  word not in string.punctuation): # remove punctuation
            stem_word = stemmer.stem(word) # stemming word
            tweets_clean.append(stem_word)

    return tweets_clean

df['tweet_clean'] = df['remove_user'].apply(lambda x: clean_tweets(x))
df
```

Gambar 4.18 Proses *Tokenizing*, *Stopword Removal* dan *Stemming*

Tahap berikutnya yang dilakukan merupakan pembersihan tanda baca seperti koma (,) dan kurung siku ([]) pada *tweet_clean* dan menampilkannya pada kolom baru untuk perbandingan seperti Gambar 4.19 di bawah ini. Untuk penulisan kodenya dapat terlihat pada Gambar 4.20.

	Tweet	Label	translate	remove_user	tweet_clean	tweet_final
0	waspada! dugaan kebocoran data pribadi terjadi...	0	waspada! dugaan kebocoran data pribadi terjadi...	waspada! dugaan kebocoran data pribadi terjadi...	[waspada, duga, bocor, data, pribadi, situs, b...	waspada duga bocor data pribadi situs breached...
1	berdasarkan pengamatan atas penggalan data yan...	1	berdasarkan pengamatan atas penggalan data yan...	berdasarkan pengamatan atas penggalan data yan...	[dasar, amat, penggal, data, sebar, akun, bjor...	dasar amat penggal data sebar akun bjorka simp...
2	berdasarkan pengamatan atas penggalan data yan...	1	berdasarkan pengamatan atas penggalan data yan...	berdasarkan pengamatan atas penggalan data yan...	[dasar, amat, penggal, data, sebar, akun, bjor...	dasar amat penggal data sebar akun bjorka simp...
3	kominfo berkelit soal tuduhan akun bjorka yang...	0	kominfo berkelit soal tuduhan akun bjorka yang...	kominfo berkelit soal tuduhan akun bjorka yang...	[kominfo, kelit, tuduh, akun, bjorka, duga, bo...	kominfo kelit tuduh akun bjorka duga bocor dat...
4	berdasarkan pengamatan atas penggalan data yan...	0	berdasarkan pengamatan atas penggalan data yan...	berdasarkan pengamatan atas penggalan data yan...	[dasar, amat, penggal, data, sebar, akun, bjor...	dasar amat penggal data sebar akun bjorka simp...
5	kominfo berkelit soal tuduhan akun bjorka yang...	0	kominfo berkelit soal tuduhan akun bjorka yang...	kominfo berkelit soal tuduhan akun bjorka yang...	[kominfo, kelit, tuduh, akun, bjorka, duga, bo...	kominfo kelit tuduh akun bjorka duga bocor dat...
6	kominfo berkelit soal tuduhan akun bjorka yang...	0	kominfo berkelit soal tuduhan akun bjorka yang...	kominfo berkelit soal tuduhan akun bjorka yang...	[kominfo, kelit, tuduh, akun, bjorka, duga, bo...	kominfo kelit tuduh akun bjorka duga bocor dat...
7	2.dan dijual di sebuah forum online "breached ...	0	2.dan dijual di sebuah forum online "breached ...	2.dan dijual di sebuah forum online "breached ...	[jual, forum, online, breached, forums, duga, ...	jual forum online breached forums duga bocor d...
8	5.nomor hp yang dibagikan bjorka merupakan asl...	1	5.nomor hp yang dibagikan bjorka merupakan asl...	5.nomor hp yang dibagikan bjorka merupakan asl...	[nomor, hp, bagi, bjorka, asli, milik, kompast...	nomor hp bagi bjorka asli milik kompastekno us...
9	kominfo berkelit soal tuduhan akun bjorka yang...	0	kominfo berkelit soal tuduhan akun bjorka yang...	kominfo berkelit soal tuduhan akun bjorka yang...	[kominfo, kelit, tuduh, akun, bjorka, duga, bo...	kominfo kelit tuduh akun bjorka duga bocor dat...

Gambar 4.19 Hasil membersihkan *tweet_clean*

```
#remove punct di tweet final
def remove_punct(text):
    text  = " ".join([char for char in text if char not in string.punctuation])
    return text
df['tweet_final'] = df['tweet_clean'].apply(lambda x: remove_punct(x))
df.head(10)
```

Gambar 4.20 Proses membersihkan *tweet_clean*

Kemudian pada tahap berikutnya adalah menghapus kolom yang tidak akan digunakan pada tahap selanjutnya yaitu kolom *tweet*, *translate*, *remove_user* dan *tweet_clean*, sehingga hanya menyisakan 2 kolom yaitu Label dan *tweet_final* seperti Gambar 4.21 di bawah ini. Untuk penulisan kodenya dapat dilihat pada Gambar L1.3 pada lampiran.

	Label	tweet_final
0	0	waspada duga bocor data pribadi situs breached...
1	1	dasar amat penggal data sebar akun bjorka simp...
2	1	dasar amat penggal data sebar akun bjorka simp...
3	0	kominfo kelit tuduh akun bjorka duga bocor dat...
4	0	dasar amat penggal data sebar akun bjorka simp...
...	...	...
1295	0	indonesia gak abis abis yah brigadir j sambo b...
1296	0	maling ribut ribut bocor data bjorka bossman fa
1297	0	tau gak sih indonesia posisi daftar negara boc...
1298	0	tanya bentuk lindung data susah payah kumpul w...
1299	0	yg diblock mah akun platform mbok ya mikir nap...

1300 rows × 2 columns

Gambar 4.21 Hasil proses penghapusan kolom yang tidak digunakan

Tahap berikutnya yang terakhir yaitu mencari *tweet* yang kosong kemudian dihapus duplikatnya menggunakan kode seperti pada Gambar L1.4 dan Gambar L1.5 pada lampiran. Proses penghapusan *tweet* kosong dapat dilihat pada Gambar

4.22 dan hasilnya dapat dilihat lagi pada Gambar 4.23. Dari hasil proses tersebut, dapat diketahui bahwa data berkurang menjadi 1017 data yang akan dilakukan pembagian data untuk dilanjutkan ke proses pembobotan TF-IDF.

	Label	tweet_final
550	1	
1015	1	
355	1	
495	0	
23	0	
...	...	...
1182	0	yg ngikutin bjorka bocor data cari suara yg wa...
89	0	yg pancing bales si bjorka kominfo bocor data ...
250	0	yg si bjorka ttg bocor data elu elu idola atuh...
886	0	yudhistira nugraha s t m ict adv d phil kepala...
875	0	yudhistira nugraha s t m ict adv d phil kepala...

1300 rows × 2 columns

Gambar 4.22 Sebelum proses penghapusan duplikat yang kosong

	Label	tweet_final
550	1	
120	0	agan bjorka personal indonesia gimana sih wkwk...
420	0	ajar hacker bjorka bocor data ict juluk negara...
1275	0	ajar hacker bjorka bocor data ict juluk negara...
682	0	aksi curi data hacker bjorka nilai bssn serang...
...	...	...
1182	0	yg ngikutin bjorka bocor data cari suara yg wa...
89	0	yg pancing bales si bjorka kominfo bocor data ...
250	0	yg si bjorka ttg bocor data elu elu idola atuh...
886	0	yudhistira nugraha s t m ict adv d phil kepala...
875	0	yudhistira nugraha s t m ict adv d phil kepala...

1017 rows × 2 columns

Gambar 4.23 Hasil proses penghapusan duplikat yang kosong

4.4 Pembagian Data

Text pre-processing menghasilkan 1017 data yang tersisa untuk dilakukan pembagian data, kemudian data tersebut akan dibagi menjadi 80% *data training* dan 20% *data testing* atau sebanyak 813 *data training* dan 204 *data testing*. Dalam pembagian data juga ditambahkan parameter untuk menduplikat kolom yaitu *copy_data_test* yang digunakan beberapa proses ke depan seperti *export* ke *file excel*, kemudian dilakukan juga pendefinisian untuk *tweet_final* yaitu *ft_data_test* yang akan digunakan untuk proses validasi, evaluasi, visualisasi dan lainnya. Untuk penulisan kodenya dapat dilihat pada Gambar 4.26. Seperti yang terlihat pada Gambar 4.24 di bawah ini, kolom *tweet_final* pada *dataframe* dapat dibagi dengan baik dan sesuai, sedangkan untuk kolom label dapat dilihat pada Gambar 4.25.

```
(75     dear kaya emang u dah nganunya rakyat korban t...
 918    pakar komunikasi unair tips atas bocor data un...
 530    perintah salah bocor data duduk security nya k...
 202    bjorka sorot netizen sosok gunjing klaim bobol...
 903    mas talk episode hallo teman teman informasi s...
                                   ...
 405    mahfud md bocor data retas bjorka sifat bahaya...
 639    kait bjorka bilang fenomena gunung es bocor da...
 200    bjorka trending bocor data ya ngaruh rakyat yg...
 452    indonesia negara korban bocor data asia tengga...
 568    naskah berita transkrip wawancara kutip parafr...
Name: tweet_final, Length: 813, dtype: object,
 502     presiden bentuk tim khusus tangan bocor data p...
 1111    gak mudeng dgn pola pikir asong nawacita jokow...
 205     bssn telusur duga insiden bocor data yg valida...
 539     dasar temu akun pantau bocor data dark tracer ...
 605          akun twitter tangguh kontroversi bocor data
                                   ...
 578     data pribadi mmg gak krn mahfud jabat negara b...
 893     lhopresiden jokowi instruksi jajar tindak lanj...
 799     langsung bentuk tim khusus tanggap bocor data ...
 956     mah otak bjorka tangkap selesai cenderung alih...
 919     pakar komunikasi unair tips atas bocor data un...
Name: tweet_final, Length: 204, dtype: object,
```

Gambar 4.24 *Data Training* dan *Data Test tweet_final*

```
75     0
918    0
530    0
202    0
903    0
      ..
405    0
639    0
200    0
452    0
568    0
Name: Label, Length: 813, dtype: int64,
502     0
1111    0
205     0
539     0
605     0
       ..
578     0
893     0
799     0
956     0
919     0
Name: Label, Length: 204, dtype: int64)
```

Gambar 4.25 *Data Training* dan *Data Test* Label

```
#pembagian data
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(df['tweet_final'], df['Label'], train_size=0.8, stratify=df['Label'], random_state=42)

copy_data_test = X_test.copy()
ft_data_test = df['tweet_final']

X_train, X_test, y_train, y_test
# pd.DataFrame(X_train)
# pd.DataFrame(y_train)
# pd.DataFrame(X_test)
# pd.DataFrame(y_test)
```

Gambar 4.26 Proses pembagian data

4.5 Pembobotan TF-IDF

Pembobotan TF-IDF dilakukan dengan melakukan *import TfidfVectorizer* melalui *library sklearn*, sehingga diperlukan *install library sklearn* terlebih dahulu. Proses TF-IDF ini bertujuan untuk menghitung bobot setiap kata, sehingga semakin besar bobot suatu kata tersebut maka kata tersebut semakin penting. Pembobotan TF-IDF dilakukan pada 3 *data frame* (yang diberi bobot merupakan setiap kata pada *tweet* atau cuitan), yaitu *X_train* atau *data training* kolom *tweet_final*, *X_test* atau *data testing* kolom *tweet_final* dan *f_test* atau seluruh data pada kolom *tweet_final* (1017 *tweet*). Penulisan kodenya dapat dilihat pada Gambar 4.27, sedangkan hasilnya dapat dilihat pada Gambar 4.28.

```
#pembobotan tf idf
from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = TfidfVectorizer()

x_train = vectorizer.fit_transform(X_train)
x_test = vectorizer.transform(X_test)
f_test = vectorizer.transform(ft_data_test)

# print(x_train)
print(x_test)
# print(f_test)
```

Gambar 4.27 Proses pembobotan TF-IDF

```
(0, 2074)     0.4542202093887527
(0, 1980)     0.2620185458068892
(0, 1635)     0.2079923257435548
(0, 1633)     0.2137027731266956
(0, 1597)     0.2692474086592812
(0, 1505)     0.37238351273989834
(0, 1383)     0.1837837465870749
(0, 1289)     0.22711010469437634
(0, 1011)     0.24620119802559348
(0, 821)      0.3183572926765639
(0, 791)      0.3393875163184614
(0, 404)      0.0645815424659846
(0, 274)      0.06853577528061967
(0, 250)      0.07177577279138407
(0, 207)      0.22002617073636196
(1, 2249)     0.33687719800429244
(1, 2178)     0.2036499439211254
(1, 2134)     0.19645463440406616
(1, 2004)     0.2129263112712394
(1, 1803)     0.4258526225424788
(1, 1777)     0.22600061673076866
(1, 1606)     0.22600061673076866
(1, 1588)     0.19645463440406616
(1, 1379)     0.22600061673076866
(1, 1338)     0.22600061673076866
```

Gambar 4.28 *Data Testing* pembobotan TF-IDF

Proses pembobotan TF-IDF yaitu menghitung nilai (bobot) setiap kata dengan tujuan untuk mengetahui seberapa penting sebuah kata di dalam kalimat (sekelompok kata). Semakin sering suatu kata muncul dalam dokumen maka semakin besar bobot kata tersebut. Gambar 4.28 menunjukkan *data testing* yang sudah dilakukan pembobotan.

4.6 Klasifikasi *Support Vector Machine*

Setelah melakukan pembobotan maka dilakukan klasifikasi menggunakan algoritma *Support Vector Machine*. Klasifikasi SVM ini merupakan pengolahan data yang disebut *supervised learning* untuk memprediksi label berdasarkan pelatihan yang sudah dilakukan dengan *data training* yang sudah berlabel. Proses pertama yang dilakukan merupakan melatih algoritma dengan *data training* untuk mengenali sebuah data, kemudian dilanjutkan dengan melakukan prediksi pada *data testing* kolom *tweet_final* (*x_test*) yang akan diberi nama *predict* dan *data full* kolom *tweet_final* (*f_test*) yang diberi nama *f_test_predict*, kemudian ditampilkan hasilnya berupa angka yang jika didefinisikan negatif = 0 dan positif = 1 dengan menampilkan juga panjang data sesuai dengan *data testing*. Untuk penulisan kodenya dapat dilihat pada Gambar 4.29, sedangkan hasilnya dapat dilihat pada Gambar 4.30.

```
#svm
from sklearn import svm
import sys

clf = svm.SVC(kernel="linear")
clf.fit(x_train, y_train)
predict = clf.predict(x_test)
f_test_predict = clf.predict(f_test)
np.set_printoptions(threshold=sys.maxsize)

print(predict)
len(predict)
```

Gambar 4.29 Proses *Support Vector Machine*

```
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 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 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 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 0 0 0]
204
```

Gambar 4.30 Proses *Support Vector Machine*

Klasifikasi SVM merupakan klasifikasi yang menggunakan *supervised learning* (memprediksi kategori berdasarkan model atau pola dari proses *training*) sehingga *data training* akan menjadi digunakan sebagai parameter oleh SVM untuk melakukan pelabelan pada *data testing*. Proses prediksi SVM berhasil dilakukan oleh komputer, dengan melakukan *training* terlebih dahulu menggunakan *clf.fit(x_train,y_train)* kemudian dilakukan prediksi (perubahan label) pada data lengkap (*f_test_predict*) dan *data testing* (*predict*). Pada Gambar 4.30 menunjukkan hasil prediksi SVM dengan rincian 202 label negatif dan 2 label negatif pada *data testing*.

Langkah selanjutnya yaitu melakukan pengubahan angka pada label menjadi kata negatif atau positif, sehingga ketika melakukan *export to excel* data menjadi lebih jelas dengan menampilkan kolom *tweet_final* (dari *copy_data_test*) dan Label (dari *predict*), Gambar 4.31 di bawah ini merupakan hasil dari pengubahan angka Label menjadi kata negatif dan positif. Untuk penulisan kodenya dapat dilihat pada Gambar L1.6 pada lampiran.

	Label
0	NEGATIF
1	NEGATIF
2	NEGATIF
3	NEGATIF
4	NEGATIF
...	...
199	NEGATIF
200	NEGATIF
201	NEGATIF
202	NEGATIF
203	NEGATIF

204 rows × 1 columns

Gambar 4.31 Hasil pengubahan Label

Berikutnya yaitu menyimpan hasil *data testing* ke *excel* untuk dilanjutkan ke proses validasi dan evaluasi data, karena hasil *data testing* tersebut memiliki dua kolom yang tidak digunakan, sehingga melakukan *export to excel* akan mempermudah proses validasi menggunakan *operator drop(columns)* seperti pada Gambar L1.7 pada lampiran. Gambar 4.32 di bawah ini merupakan hasil *data testing* pada *file excel* dan Gambar 4.33 merupakan hasil *data testing* ketika sudah dihapus kolom yang tidak digunakan.

1		tweet_final		Label
2	502	presiden bentuk tim khusus tangan bocor data pribadi institusi bjorka pembetukan tim menkominfo jhonny g plate panggil istana ne	0	NEGATIF
3	1111	gak mudeng dgn pola pikir asong nawacita jokowi sbg yg mengatasnamakan masyarakat yg tuntut mesti selenggara mang data yg tdk	1	NEGATIF
4	205	bssn telusur duga insiden bocor data yg validasi thdp data yg dip	2	NEGATIF
5	539	dasar temu akun pantau bocor data dark tracer retas kelas bjorka coba retas target indonesia bocor data	3	NEGATIF
6	605	akun twitter tangguh kontroversi bocor data	4	NEGATIF
7	219	apa parah banget si kelas presiden ketua dpr ri data pribadi bocor s	5	NEGATIF
8	293	fakta aman data indonesia rentan bobol aku jadi bocor data ka	6	NEGATIF
9	800	populer badan siber sandi negara bssn usut aksi hacker bjorka aku lemah sistem faktor manusia bocor data	7	NEGATIF
10	367	bocor data bin pasti dokumen aman bjorkanism bjorka	8	NEGATIF
11	515	bocor data bjorka johnny g plate spesifik mahfud md pasti data rahasia	9	NEGATIF
12	344	bocor data parah sikap perintah lempar tanggung	10	NEGATIF
13	544	bocor data bin pasti dokumen aman bjorka	11	NEGATIF
14	963	polri tetap orang sangka bocor data perintah hacker retas bjorka bjorka kasusperetas polri updatebali	12	NEGATIF
15	609	rapat bocor data lupa hacker bjorka bahas bahas rencana strategi aman menkominfo johnny g plate data yg edar publik data data	13	NEGATIF
16	26	dasar amat penggal data sebar akun bjorka simpul data asal	14	NEGATIF
17	1040	berita bjorka bikin heboh retas bocor data situs instansi kpu bssn database polri retas lemah aman siber indonesia liputansctv	15	NEGATIF
18	936	puan satgas lindung data selesai bocor data bjorka kasuskebocorandata retas p	16	NEGATIF
19	437	anggota komisi i dpr ri fadli zon kritik perintah kait bocor data hacker bjorka	17	NEGATIF
20	365	hacker bjorka klaim data sensitif duduk indonesia dapat sumber resmi	18	NEGATIF
21	1018	tarik cermat bjorka yg jd idola flashback brandal lokajaya putra tumenggung wilatikta si robinhood musuh sheriff of nottingham sama	19	NEGATIF
22	409	johnny g plate bentuk tim darurat emergency response team kait marak bocor data bjorka	20	NEGATIF

Gambar 4.32 Hasil *data testing* pada *file excel*

	tweet_final	Label
0	presiden bentuk tim khusus tangan bocor data p...	NEGATIF
1	gak mudeng dgn pola pikir asong nawacita jokow...	NEGATIF
2	bssn telusur duga insiden bocor data yg valida...	NEGATIF
3	dasar temu akun pantau bocor data dark tracer ...	NEGATIF
4	akun twitter tangguh kontroversi bocor data	NEGATIF
...	...	...
199	data pribadi mmg gak krn mahfud jabat negara b...	NEGATIF
200	lhopresiden jokowi instruksi jajar tindak lanj...	NEGATIF
201	langsung bentuk tim khusus tanggap bocor data ...	NEGATIF
202	mah otak bjorka tangkap selesai cenderung alih...	NEGATIF
203	pakar komunikasi unair tips atas bocor data un...	NEGATIF

204 rows × 2 columns

Gambar 4.33 Hasil *data testing drop* kolom

4.7 Validasi dan Evaluasi Data

Proses yang dilakukan berikutnya merupakan validasi data menggunakan *K-fold Cross Validation* untuk mengetahui seberapa baik klasifikasi yang dilakukan algoritma SVM. Dalam penelitian ini peneliti menggunakan *10-fold Cross Validation* dalam artian data akan dibagi menjadi 10 bagian sama banyak dan setiap *testing* (10-*fold*) dibagi menjadi 9 kali *training* data dan 1 kali *testing* data. Menurut Arya (2022), *K-fold* yang digunakan biasanya antara 5 sampai 10 dan tidak peraturan terkait harus menggunakan berapa *fold*, perbedaannya hanya pada semakin banyak nilai *K* maka semakin bagus nilainya. Menggunakan k = 10 lebih efisien secara komputasi karena tidak memakan waktu yang banyak dan jika lebih dari 10 membuat secara komputasi tidak praktis, jika menggunakan nilai k terlalu kecil juga lebih efisien secara komputasi tetapi akan meningkatkan kemungkinan nilai yang cukup bias. Untuk hasilnya dapat dilihat pada Tabel 4.1 di bawah ini. Untuk penulisan kodenya dapat dilihat pada Gambar 4.34.

```
#kfold cv
from sklearn.model_selection import cross_val_score
score = cross_val_score(clf, x_train, y_train, cv=10)
score = list(map('{}'.format, score))

print("K-Fold(10): "f'{score}')
print("Rata-rata: "f"{cross_val_score(clf, x_train, y_train, cv=10).mean()}")
```

Gambar 4.34 Proses *10-fold Cross Validation*

Tabel 4.1 Hasil validasi *10-fold Cross Validation*

Fold ke-	*Score*
1	92.68%
2	92.68%
3	91.46%
4	92.59%
5	92.59%
6	92.59%
7	95.06%
8	92.59%
9	91.35%
10	91.35%
Rata-rata	92.49%

Validasi menggunakan *K-fold Cross Validation* bertujuan untuk mengetahui seberapa baik klasifikasi yang dilakukan algoritma SVM. Hasil dari *10-fold Cross Validation* dapat dilihat pada Tabel 4.1 dengan rata-rata sebesar 92.49%, sehingga dapat dikatakan bahwa algoritma SVM memiliki performa yang sangat baik karena mendekati nilai sempurna (100%).

Setelah dilakukan proses validasi, selanjutnya melakukan evaluasi dengan membandingkan Label sebelum dilakukan klasifikasian oleh SVM (*y_test*) dengan Label *predict* yang sudah melalui proses SVM (hasil prediksi Label *data testing*). Pada Gambar 4.35 menjelaskan bahwa dalam proses melakukan evaluasi yang pertama dilakukan yaitu mengetahui nilai *True Negative (TN), False Negative (FN), True Postive (TP)* dan *False Positive (FP)* menggunakan *library sklearn.meatrics import confusion_matrix*, hasilnya dapat dilihat pada Tabel 4.2.

```
#confusion matrix
from sklearn.metrics import recall_score, precision_score, confusion_matrix, accuracy_score, f1_score

tn, fp, fn, tp = confusion_matrix(y_test, predict).ravel()

# F1 = 2 * (precision * recall) / (precision + recall)
# print(tn)
# print(fn)
# print(tp)
# print(fp)

print("True Negative (TN):", tn)
print("False Negative (FN):", fn)
print("True Positive (TP):", tp)
print("False Positive (FP):", fp)
print(" ")
print("F1:",f"{f1_score(y_test, predict)}")
print("Accuracy:",f"{accuracy_score(y_test, predict)}")
print("Precision:",f"{precision_score(y_test, predict)}")
print("Recall:",f"{recall_score(y_test, predict)}")
```

Gambar 4.35 Proses *Confusion Matrix*

Tabel 4.2 Hasil evaluasi *Confusion Matrix*

Class		Prediksi	
		Positif	Negatif
Aktual	Positif	*True Positive (TP)* *2*	*False Negative (FN)* *15*
	Negatif	*False Positive (FP)* 0	*True Negative (TN)* 187

Nilai *True Negative* mendapatkan 187 yang menunjukkan bahwa label negatif yang diprediksi benar oleh SVM. Nilai *False Negative* mendapatkan 15 yang menunjukkan bahwa label negatif yang diprediksi salah (SVM melakukan prediksi positif yang seharusnya menjadi prediksi negatif). Nilai *True Positive* mendapatkan 2 label positif yang diprediksi benar oleh SVM. Nilai *False Positive* mendapatkan 0 yang menunjukkan bahwa label positif yang diprediksi salah (SVM melakukan prediksi negatif yang seharusnya menjadi prediksi positif).

```
True Negative (TN): 187
False Negative (FN): 15
True Positive (TP): 2
False Positive (FP): 0

F1: 0.21052631578947367
Accuracy: 0.9264705882352942
Precision: 1.0
Recall: 0.11764705882352941
```

Gambar 4.36 Hasil evaluasi *Confusion Matrix*

Dari hasil *Confusion Matrix* maka akan dilakukan pencarian nilai *F1*, *Accuracy*, *Precision* dan *Recall* untuk mengetahui seberapa akurat algoritma SVM dalam melakukan klasifikasi.

Nilai *F1* yang dihasilkan adalah 0.21, nilai ini menunjukkan perbandingan rata-rata nilai *precision* dan *recall* untuk menunjukkan apakah SVM memiliki nilai *precision* dan *recall* yang baik. Nilai *Accuracy* yang dihasilkan adalah 0.93, nilai ini menunjukkan seberapa sesuai hasil prediksi SVM dengan label sesungguhnya. Nilai *Precision* yang dihasilkan adalah 1, nilai ini menunjukkan seberapa besar tingkat konsistensi prediksi SVM dengan label sesungguhnya. Nilai *Recall* yang dihasilkan adalah 0.12, nilai ini menunjukkan seberapa besar kesuksesan SVM dalam melakukan prediksi (rasio prediksi yang benar). Dari hasil *Confusion Matrix* dapat diketahui bahwa total label positif sebanyak 17 *tweet* dan negatif sebanyak

187 *tweet* (memiliki perbandingan yang cukup jauh dengan selisih 170), hal ini dapat disebabkan oleh kurangnya data yang di-*training* maupun faktor lain seperti saat pelabelan data secara manual terdapat perbandingan yang jauh, sehingga menurut Wijaya (2020) ketika klasifikasi positif dan negatif tidak seimbang maka *metric F1-score* lebih baik digunakan daripada nilai *Accuracy*, hal ini didukung juga oleh Balakrishnan (2019) yang menyatakan *accuracy* bukanlah *metric* yang baik ketika sekumpulan data tidak seimbang. Nilai *F1* yang dihasilkan (0.21) menunjukkan bahwa keakuratan algoritma SVM melakukan klasifikasi (pelabelan) sangat rendah.

4.8 Visualisasi Data

Dalam melakukan visualisasi data memerlukan data lengkap *tweet_final* dan Label yang sudah diproses oleh SVM, sehingga untuk menyiapkannya langkah pertama yang dilakukan oleh peneliti yaitu mengubah angka Label menjadi kata negatif dan positif lalu melakukan *export to excel* data untuk dilanjutkan ke proses visualisasi data. Gambar 4.37 di bawah ini merupakan hasil *excel* dari pelabelan yang dilakukan SVM. Untuk penulisan kode mengubah angka Label menjadi kata negatif dan positif dapat dilihat pada Gambar L1. 8 pada lampiran, sedangkan untuk melakukan *export to excel* dapat dilihat pada Gambar 4.38.

1		tweet_final		Label
2	550		0	NEGATIF
3	120	agan bjorka personal indonesia gimana sih wkwkwk kemarin a	1	NEGATIF
4	420	ajar hacker bjorka bocor data ict juluk negara open source	2	NEGATIF
5	1275	ajar hacker bjorka bocor data ict juluk negara open source pedoman tangerang	3	NEGATIF
6	682	aksi curi data hacker bjorka nilai bssn serang siber klasifikasi rendah bssn aku bocor data valid valid	4	NEGATIF
7	469	aksi retas hacker nama bjorka bobol data perintah sita perhati presiden joko widodo instruksi k	5	NEGATIF
8	454	aksi retas hacker nama bjorka bobol data perintah sita perhati presiden joko widodo instruksi khus	6	NEGATIF
9	504	aksi tambah daftar bocor data indonesia marak mo	7	NEGATIF
10	332	aksi tambah daftar bocor data indonesia marak motif bjorka bocor data platform breached koranter	8	NEGATIF
11	668	aku bocor data buru bjorka lambat	9	NEGATIF
12	768	akun bjorka grup telegram bagi data pribadi jabat publik menkopolhukam klarifikasi rahasia negara	10	NEGATIF
13	785	akun bjorka grup telegram bagi data pribadi jabat publik menkopolhukam mengklarif	11	NEGATIF
14	109	akun bjorka klaim jadi jokowi target perkara bocor data	12	NEGATIF
15	427	akun goblok kenal bjorka aja udah pinter sejagad raya kalo emang peduli bocor data join raidforum	13	NEGATIF
16	1041	akun telegram jual sangka bocor data perintah nama	14	NEGATIF
17	605	akun twitter tangguh kontroversi bocor data	15	NEGATIF

Gambar 4.37 Hasil *excel* untuk visualisasi data

```
#buat file excel
writer = pd.ExcelWriter('HasilDataFull.xlsx',engine='xlsxwriter')
workbook = writer.book
worksheet = workbook.add_worksheet('Validation')
writer.sheets['Validation'] = worksheet
all_data.to_excel(writer,sheet_name='Validation',startrow=0 , startcol=0)
all_label_test.to_excel(writer,sheet_name='Validation',startrow=0, startcol=2)
writer.save()

drop_kolom = pd.read_excel("HasilDataFull.xlsx")
hdf = drop_kolom.drop(columns=['Unnamed: 0', 'Unnamed: 2'])
hdf["tweet_final"] = hdf["tweet_final"].astype(str)
hdf
```

Gambar 4.38 *Export to Excel* dan *drop* kolom untuk visualisasi

Seperti yang terlihat pada *excel* di atas menunjukkan bahwa terdapat kolom yang tidak digunakan sehingga akan dilakukan proses *drop(columns)* untuk menghapus kolom yang tidak digunakan seperti Gambar 4.39 di bawah ini, sedangkan untuk penulisan kodenya dapat dilihat pada Gambar 4.38.

	tweet_final	Label
0	nan	NEGATIF
1	agan bjorka personal indonesia gimana sih wkwk...	NEGATIF
2	ajar hacker bjorka bocor data ict juluk negara...	NEGATIF
3	ajar hacker bjorka bocor data ict juluk negara...	NEGATIF
4	aksi curi data hacker bjorka nilai bssn serang...	NEGATIF
...	...	...
1012	yg ngikutin bjorka bocor data cari suara yg wa...	NEGATIF
1013	yg pancing bales si bjorka kominfo bocor data ...	NEGATIF
1014	yg si bjorka ttg bocor data elu elu idola atuh...	NEGATIF
1015	yudhistira nugraha s t m ict adv d phil kepala...	NEGATIF
1016	yudhistira nugraha s t m ict adv d phil kepala...	NEGATIF

1017 rows × 2 columns

Gambar 4.39 Hasil *excel* setelah diunggah dan *drop* kolom

Langkah selanjutnya yaitu membuat *WordCloud* menggunakan *library wordcloud* dan *matplotlib.pyplot* untuk mengatur besar atau bentuk gambar *wordcloud*, sedangkan *wordcloud* untuk memanggil kata yang sering muncul berdasarkan label positif. Ditambahkan juga *stopword* untuk menghapus kata kunci yang digunakan pencarian (karena kata tersebut pasti akan sering muncul), kata kunci tersebut adalah bocor, data dan bjorka. Pada *wordcloud* positif terdapat beberapa kata positif yang sering muncul seperti “aman”, “jamin”, dan “tenang”, sedangkan untuk *wordcloud* negatif terdapat “retas”, “duga” dan “isu”. Gambar 4.40 di bawah ini merupakan hasil *wordcloud* positif, sedangkan Gambar 4.41

merupakan hasil *wordcloud* negatif. Untuk penulisan kodenya dapat dilihat pada Gambar 4.42 untuk *wordcloud* positif sedangkan Gambar L1.9 pada lampiran untuk *wordcloud* negatif.

Gambar 4.40 *Word Cloud* Positif

Gambar 4.41 *Word Cloud* Negatif

```
#word cloud positif
from wordcloud import WordCloud
import matplotlib.pyplot as mplt

l1 = hdf[hdf["Label"]=="POSITIF"]
tweet_l1 = ' '.join(word for word in l1["tweet_final"])
stop_word = [ "bocor","data", "bjorka"]
wc = WordCloud(stopwords = stop_word, colormap='Greens', width=1000, height=1000, background_color='black', mode='RGBA').generate(tweet_l1)

mplt.figure(figsize=(10,10))
mplt.imshow(wc, interpolation='bilinear')
mplt.axis("off")
mplt.margins(x=0, y=0)
mplt.show()
```

Gambar 4.42 Proses *WordCloud* positif

Pada tahap visualisasi data, penelitian ini juga menggunakan diagram *pie chart*. Untuk menggunakan diagram *pie chart* menggunakan *library matplotlib* dengan menghitung jumlah Label masing-masing dan membandingkannya, kemudian dibentuk persentase dan menghasilkan *pie chart* pada Gambar 4.44. Untuk penulisan kodenya dapat dilihat pada Gambar 4.43.

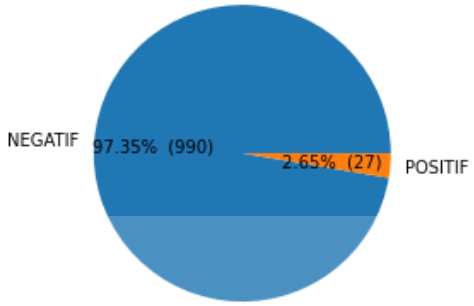
```
import matplotlib.ticker as ticker
import matplotlib.cm as cm
import matplotlib as mpl
from matplotlib.gridspec import GridSpec

title = hdf.groupby(['Label'])['Label'].count()
label = ['NEGATIF','POSITIF']

def make_autopct(title):
    def my_autopct(pct):
        total = sum(title)
        val = int(round(pct*total/100.0))
        return '{p:.2f}%  ({v:d})'.format(p=pct,v=val)
    return my_autopct

print(title)
mplt.pie(title, labels=label, autopct=make_autopct(title))
mplt.show()
```

Gambar 4.43 Proses *Pie Chart*



Gambar 4.44 Hasil *Pie Chart*

Pada Gambar 4.44 di atas, menunjukkan persentase sentimen yang sudah dilakukan klasifikasi oleh SVM. Persentase sentimen *tweet* Kebocoran Data Bjorka sebanyak 97.35% negatif atau sebanyak 990 *tweet* dan 2.65% positif atau 27 *tweet*. Hasil ini menunjukkan bahwa sentimen masyarakat terhadap Bjorka lebih banyak ke sentimen negatif, sehingga hasil ini menyatakan bahwa masyarakat kebanyakan sudah cukup memahami bahwa tindakan yang dilakukan Bjorka adalah tindakan yang negatif. Dari hasil ini juga pemerintahan dapat menjadi bahan evaluasi atau rencana strategis pemerintahan dalam menangani insiden kebocoran data ke depannya maupun meningkatkan kesadaran masyarakat dalam insiden kebocoran data dan meningkatkan keamanan data pemerintahan. Dari hasil sentimen negatif juga menyatakan bahwa pemerintahan tidak perlu terlalu krusial untuk melakukan edukasi terhadap kebocoran data yang dilakukan Bjorka, sehingga pemerintahan bisa lebih fokus untuk menangani sektor yang lain seperti meningkatkan keamanan data itu sendiri.

4.9 Hasil dan Pembahasan

Dari proses analisis data di atas, mulai dari *crawling data* hingga visualisasi data dapat disimpulkan bahwa:

1. Saat proses *crawling data* sebaiknya jika dilakukan hari itu juga saat kata kunci yang diinginkan sedang sering dibahas, sehingga tidak terdapat keterbatasan data yang dapat mengganggu proses penelitian atau dapat dilakukan perubahan pada kata kunci agar mendapatkan data yang lebih banyak maupun yang lebih spesifik pada suatu penelitian.
2. Dalam melakukan pelabelan manual, disarankan dilakukannya penyelarasan pemahaman dalam melakukan pelabelan oleh sukarelawan. Hal ini untuk meningkatkan nilai reliabilitas yang tinggi agar dapat menghasilkan data yang reliabel.
3. Pada proses *text pre-processing* bagian *translation* terdapat masalah dengan penerjemahan Bahasa Inggris, sehingga beberapa kata dalam bahasa inggris masih lolos dari proses tersebut. Hal ini dapat disebabkan karena terdapat kata Bahasa Inggris yang tercampur imbuhan Bahasa Indonesia maupun faktor-faktor lainnya. Pada bagian *stopword removal* juga ditemukan permasalahan seperti penggunaan *library stopwords* yang kurang tepat, sebagai contoh kalimat positif "BIN mengatakan tidak ada kejadian kebocoran data" menjadi kalimat negatif "BIN mengatakan ada kejadian kebocoran data".
4. Dari hasil validasi menggunakan 10-*fold Cross Validation* mendapatkan rata-rata nilai sebesar 92.49%, nilai tersebut menyatakan bahwa algoritma SVM memiliki performa yang sangat baik karena mendekati nilai sempurna (100%).
5. Dari hasil evaluasi menggunakan *Confusion Matrix* menyatakan saat SVM melakukan klasifikasi terhadap *data testing* terdapat 17 label positif dan 187 label negatif, sehingga terdapat perbandingan yang jauh antar label yang membuat nilai *F1-score* lebih baik digunakan daripada nilai *Accuracy*. Nilai *F1-Score* yang dihasilkan adalah 0.21, nilai tersebut menunjukkan bahwa keakuratan algoritma SVM dalam melakukan klasifikasi sangat rendah. Hal ini dapat disebabkan karena kesalahan saat melakukan pelabelan data secara manual dan juga dapat disebabkan dengan berbagai faktor lainnya seperti

penggunaan *library stopwords* yang kurang tepat maupun terdapat beberapa kata yang tidak diketahui oleh sistem seperti bahasa gaul atau kata singkatan, sehingga algoritma SVM melakukan kesalahan dalam melakukan klasifikasi (pelabelan).

6. Berdasarkan visualisasi data menggunakan *wordcloud* terdapat kata positif yang sering muncul seperti “aman” , “jamin” dan “tenang” sedangkan kata negatif terdapat “retas”, “duga” dan “isu”.
7. Pada proses visualisasi data bagian *pie chart*, ditemukan sebanyak 97.35% negatif atau sebanyak 990 *tweet* dan 2.65% positif atau 27 *tweet*, sehingga dapat diketahui tanggapan publik lebih banyak beranggapan negatif terhadap kebocoran data yang dilakukan oleh Bjorka. Dari hasil sentimen tersebut juga menyatakan bahwa edukasi ke masyarakat terhadap kebocoran data yang dilakukan oleh Bjorka tidak terlalu krusial, pemerintahan bisa lebih fokus untuk menangani sektor yang lain seperti meningkatkan keamanan data itu sendiri maupun menyiapkan edukasi kebocoran data lainnya.

BAB V
PENUTUP

5.1 Kesimpulan

Berdasarkan proses yang telah dilakukan di atas dapat ditarik kesimpulan bahwa:

1. Dari 1017 *tweet* yang berisi tanggapan masyarakat, 2.65% sentimen positif masyarakat adalah masyarakat yang mendukung aksi Bjorka dalam melakukan pembocoran data sedangkan 97.35% sentimen negatif masyarakat lainnya adalah masyarakat yang tidak mendukung aksi Bjorka dalam melakukan pembocoran data. Maka, pemerintah tidak perlu terlalu memfokuskan untuk melakukan edukasi ke masyarakat terhadap aksi pembocoran data oleh Bjorka dan lebih meningkatkan keamanan data agar tidak terjadi insiden yang sama.
2. Dari hasil validasi dan evaluasi menunjukkan bahwa algoritma SVM dapat melakukan klasifikasi dengan baik dengan rata-rata nilai validasi sebesar 92.49% , namun dengan tingkat keakuratan yang rendah yaitu *F1* = 20% maka sebaiknya menggunakan data yang lebih banyak maupun menggunakan data yang seimbang nilai negatif dan positifnya atau mencoba menggunakan algoritma selain SVM pada data yang tidak imbang.

5.2 Saran

Saran yang dapat dilakukan untuk penelitian selanjutnya yaitu:

1. Pada penelitian selanjutnya diharapkan untuk menggunakan data yang lebih banyak.
2. Menambah jumlah *data training* yang digunakan.
3. Menggunakan sukarelawan yang memiliki pemahaman cukup pada pelabelan.
4. Diharapkan dapat melakukan *crawling data* pada waktu dengan kondisi yang dibutuhkan (waktu yang tepat).
5. Menggunakan *library stopwords* yang lebih terkini/*update* terhadap bahasa gaul dan dapat melakukan eleminasi kata-kata yang disingkat secara manual maupun mencari yang tersedia di *python*.
6. Menggunakan kamus positif dan negatif yang berbeda saat proses pelabelan manual dengan kata yang lebih terkini.
7. Menggunakan algoritma yang berbeda dengan *Support Vector Machine*.
8. Mengganti kata kunci pencarian.
9. Menggunakan *library translation* yang terbaru dan memastikan semua data dapat diterjemahkan dengan baik.

DAFTAR PUSTAKA

Achsanty, R. A. (2021). Voting dan Fandom K-Pop (Analisis Komunikasi Antar Penggemar TREASURE dalam Ajakan Voting di Twitter).

Aprillia, J. (2022). *Mengenal Pentingnya Keamanan Data dan Cara Menjaganya*. Diakses pada 20 September 2022, dari https://www.dewaweb.com/blog/mengenal-keamanan-data/#:~:text=Keamanan%20data%20sangat%20diperlukan%20dalam,untuk%20melindungi%20ekosistem%20teknologi%20informasi.

Balakrishnan, H. N. (2019). *Confusion Matrix, Accuracy, Precision, Recall, F1 Score*. Diakses pada 1 Desember 2022, dari https://medium.com/analytics-vidhya/confusion-matrix-accuracy-precision-recall-f1-score-ade299cf63cd

CNN Indonesia. (2022). *Dekan FHUI: Kebocoran Data Warga Tanggung Jawab Negara*. Diakses pada 20 September 2022, dari https://www.cnnindonesia.com/nasional/20220912182129-20-846811/dekan-fhui-kebocoran-data-warga-tanggung-jawab-negara

CNN Indonesia. (2022). *Kisah Kekaguman Netizen pada Bjorka, Ada Apa?* Diakses pada 20 September 2022, dari https://www.cnnindonesia.com/teknologi/20220915041428-192-848029/kisah-kekaguman-netizen-pada-bjorka-ada-apa

CNN Indonesia. (2022). *Miliaran Data SIM Card Diduga Bocor, Registrasi Nomor Hp Masih Aman?* Diakses pada 20 September 2022, dari https://www.cnnindonesia.com/teknologi/20220901124009-192-841875/miliaran-data-sim-card-diduga-bocor-registrasi-nomor-hp-masih-aman

Devid. (2017). *ID-OpinionWords*. Diakses pada 20 September 2022, dari https://github.com/masdevid/ID-OpinionWords

Fahmi, I. (2022). Diakses pada 20 Oktober 2022, dari https://twitter.com/ismailfahmi/status/1568856147612045312

Indraloka, D. S., & Santosa, B. (2017). Penerapan Text Mining untuk Melakukan Clustering Data Tweet Shopee Indonesia. *Jurnal Sains dan Seni ITS*, 52.

Jonizar. (2020). *Data, Data Mining dan Big Data*. Diakses pada 20 September 2022, dari https://www.linkedin.com/pulse/data-mining-dan-big-jonizar-aaij/?originalSubdomain=id

Kementerian Komunikasi dan Informatika Republik indonesia. (n.d.). *Profil Kementerian Komunikasi dan Informatika*. Diakses pada 20 September 2022, dari https://www.kominfo.go.id/profil

Latuny, W., Lawalata, V. O., Pailin, D. B., & Ohoirenan, R. (2021). Sentiment Analysis of Consumers for Determining the Packaging Features of Eucalyptus Oil Products. *Jurnal Ilmiah Teknik Industri*, 71-80.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.

Liu, Bing, Hu, Minqing, & Cheng, J. (2005). *Opinion Observer: Analyzing and Comparing Opinions on the Web." Proceedings of the 14th International World Wide Web Conference (WWW-2005).* Chiba.

Maulida, L. (2022). *Kominfo Bantah Kecolongan 1,3 Miliar Data Registrasi SIM Prabayar*. Diakses pada 20 September 2022, dari https://tekno.kompas.com/read/2022/09/01/15013017/kominfo-bantah-kecolongan-13-miliar-data-registrasi-sim-prabayar?page=all

Mesak, E., Kunang, Y. N., & Andryani, R. (2017). Eksplorasi Trending Topik Twitter menggunakan Text Mining.

Mukminin, A. (2021). Analisis Sentimen Publik terhadap Pelayanan Tes Swab-PCR Covid-19 di Indonesia Menggunakan Algoritma Support Vector Machine.

Negara, S. E., Andryani, R., & Saksono, P. H. (2016). Analisis Data Twitter: Ekstraksi dan Analisis Data Geospasial. *Jurnal Informatika, Sistem Kendali, dan Komputer*, 29-30.

Nugroho, A. S., Witarto, A. B., & Handoko, D. (2003). Support Vector machine Teori dan Aplikasinya dalam Bioinformatika. *Journal of Intelligent Systems*.

Pravina, A. M., Cholissodin, I., & Adikara, P. P. (2019). Analisis Sentimen Tentang opini Maskapai Penerbangan pada Dokumen Twitter Menggunakan Algoritme Support Vector Machine (SVM). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.

Raschka, S. (2016). *How to Select Support Vector Machine Kernels*. Diakses pada 2 Januari 2023, dari https://www.kdnuggets.com/2016/06/select-support-vector-machine-kernels.html

Taufik, P. S. (2018). Analisis Sentimen terhadap Tokoh Publik menggunakan Algoritma Support Vector Machine.

Trivusi. (2022). *Apa itu Kernel Trick? Pengertian dan Jenis-jenis Fungsi Kernel SVM*. Diakses pada 2 Januari 2023, dari https://www.trivusi.web.id/2022/04/fungsi-kernel-svm.html

Wibowo, N. I., Maulana, T. A., Muhammad, H., & Rakhmawati, N. A. (2021). Perbandingan Algoritma Klasifikasi Sentimen Twitter Terhadap Insiden Kebocoran Data Tokopedia.

Wijaya, V. (2020). *Validitas Rapid Test Covid 19 : Accuracy vs F1-Score, Pilih yang Mana?* Diakses pada 20 September 2022, dari https://www.teknologi-bigdata.com/2020/05/validitas-rapid-test-covid-19-akurasi-accuracy-vs-f1-score.html#:~:text=Adalah%20nilai%20Harmonic%20Mean%20(Rata,Harmonik)%20dari%20Precision%20dan%20Recall.

Zalyhaty, L. Q. (2021). Analisis Sentimen Tanggapan Masyarakat terhadap Vaksin COVID-19 menggunakan Algoritma Support Vector Machine (SVM).