JISKa, Vol. 6, No. 2, MEI, 2021, Pp. 120 – 129
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Perbandingan Algoritma Klasifikasi Sentimen Twitter Terhadap Insiden Kebocoran Data Tokopedia

Nadhif Ikbar Wibowo [(1)], Tri Andika Maulana [(2)], Hamzah Muhammad [(3)], Nur Aini Rakhmawati [(4)*]
Sistem Informasi, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember, Surabaya
e-mail : {nadhif.18052,tri.18052,hamzah.18052}@mhs.its.ac.id, nur.aini@is.its.ac.id
* Penulis korespondensi.
Artikel ini diajukan 7 November 2020, direvisi 10 Januari 2021, diterima 24 Januari 2021, dan dipublikasikan 3 Mei 2021.

Abstract

Public responses, posted on Twitter reacting to the Tokopedia data leak incident, were used as a data set to compare the performance of three different classifiers, trained using supervised learning modeling, to classify sentiment on the text. All tweets were classified into either positive, negative, or neutral classes. This study compares the performance of Random Forest, Support-Vector Machine, and Logistic Regression classifier. Data was scraped automatically and used to evaluate several models; the SVM-based model has the highest f1-score 0.503583. SVM is the best performing classifier.

Keywords: Model Performance Analysis, Sentiment Classification, Logistic Regression, Random Forest, Support Vector Machine

Abstrak

Twit respon masyarakat terhadap insiden kebocoran data Tokopedia di Twitter dimanfaatkan sebagai *dataset* untuk melakukan perbandingan performa tiga *classifier* berbeda dengan pemodelan *supervised learning* untuk melakukan klasifikasi sentimen pada teks. Setiap *twit* diklasifikasikan dalam salah satu dari tiga kelas, yaitu positif, negatif, atau netral. Penelitian ini membandingkan performa dari *classifier Random Forest*, *Support-Vector Machine*, dan *Logistic Regression*. Data *twit* diambil secara otomatis menggunakan *scrapper* dan digunakan untuk melakukan evaluasi model. Model *classifier Support Vector Machine* memiliki performa terbaik dengan *f1-score* sebesar 0.503583. SVM adalah *classifier* dengan performa terbaik.

Kata Kunci: Analisis Performa Model, Klasifikasi Sentimen, Logistic Regression, Random Forest, Support Vector Machine

1. PENDAHULUAN

Data merupakan catatan atas kumpulan fakta (Vardiansyah, 2008). Saat menggunakan internet, keamanan data menjadi aspek penting yang perlu diperhatikan. Khususnya ketika berurusan dengan data-data pribadi yang bersifat sensitif, kelalaian akan keamanan data pribadi dapat menimbulkan masalah-masalah yang berkaitan dengan privasi seseorang yang dapat menimbulkan berbagai macam kerugian dengan skala yang beragam. Data pribadi menjadi salah satu incaran utama penjahat siber. Tren serangan terhadap data bukan hanya sekedar pencurian, tetapi juga jual-beli data. Serangan tidak hanya ditujukan kepada individu namun juga industri atau organisasi (Librianty, 2016).

Salah satu insiden besar kebocoran data di Indonesia adalah bocornya 91 juta akun pengguna dan 7 juta akun pedagang pada *marketplace* Tokopedia yang terjadi pada bulan Mei tahun 2020. Data-data yang bocor mulai dari nama lengkap, tanggal lahir, nomor ponsel, lokasi, hingga jenis kelamin (CNN Indonesia, 2020). Insiden ini menuai berbagai respon dari masyarakat yang dapat diamati melalui cuitan-cuitan yang ditulis pada media sosial Twitter.

Pada penelitian ini, penulis memanfaatkan respon-respon masyarakat di Twitter untuk membuat model-model klasifikasi sentimen yang dapat menentukan apakah respon terhadap insiden

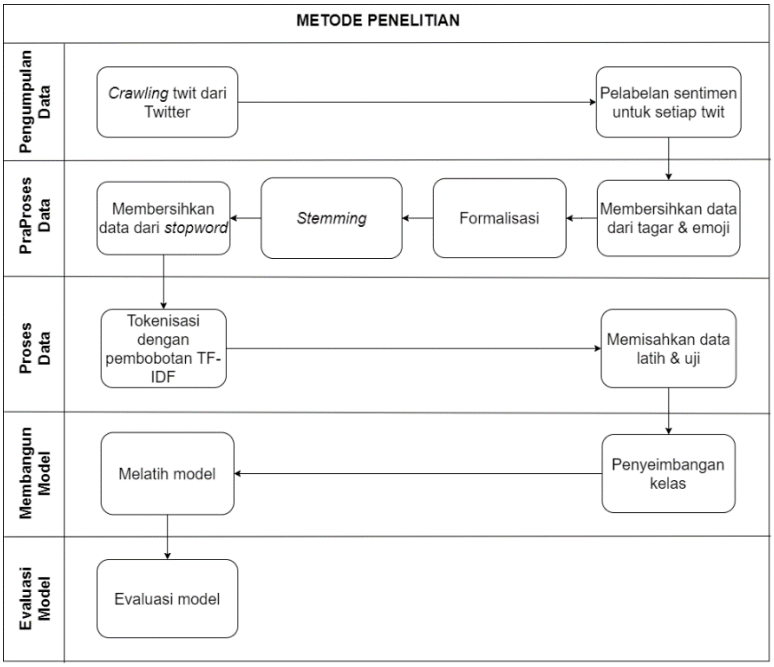
tersebut memiliki sentimen positif, negatif, atau netral. Model-model tersebut akan digunakan untuk melakukan analisis sentimen.

Masing-masing model menggunakan *classifier* yang berbeda dengan pemodelan *supervised learning*, selanjutnya model-model tersebut dibandingkan untuk mencari *classifier* terbaik. Penelitian ini membandingkan performa dari 3 *classifier* yang digunakan untuk membuat model yaitu *Random Forest, Support-Vector Machine,* dan *Logistic Regression.*

Beberapa studi sebelumnya yang membahas tentang analisis sentimen terhadap suatu topik dengan memanfaatkan data *twit* telah dilakukan. Berdasarkan (Faradhillah et al., 2016) penulis menggunakan data *twit* warga Surabaya untuk membangun model klasifikasi sentiment terhadap kinerja Pemkot Surabaya yang hasilnya ditampilkan secara interaktif dengan aplikasi berbasis web yaitu R Shiny. Didapatkan model SVM memiliki akurasi lebih baik dibanding Naïve Bayes dengan nilai akurasi 78,66% dengan menggunakan kernel RBF. Berdasarkan (Deviyanto & Wahyudi, 2018) penulis menggunakan data *twit* mengenai pemilihan gubernur DKI Jakarta tahun 2017 untuk membangun model klasifikasi sentiment menggunakan algoritma KNN. Penelitian tersebut menggunakan 2000 data *twit* dan didapatkan hasil akurasi model KNN yaitu 67,2%. Berdasarkan (Hasan et al., 2018) penelitian dilakukan analisis stentimen dengan mamanfaatkan *classifier* Naïve Beyes dengan hasil akurasi 62% pada klasifikasi sentimen akun Twitter dengan bahasa Urdu yang diterjemahkan terlebih dahulu.

2. METODE PENELITIAN

Metode penelitian secara garis besar terdiri atas 5 fase meliputi pengumpulan data, pra proses data, proses data, membangun model, dan evaluasi model. Secara lebih jelas, metode penelitian ini digambarkan oleh Gambar 1.



Gambar 1. Diagram Metode Penelitian.

2.1. Pengumpulan Data

Data diambil dengan memanfaatkan fitur *Explore* yang terdapat pada Twitter, fitur ini mendukung kueri pencarian *twit* yang sederhana, di mana hanya memanfaatkan serangkaian kata kunci, mapun kueri tingkat lanjut untuk membuat aturan pencarian yang menerapkan kondisi tertentu.

Kueri sederhana yang digunakan: “Tokopedia bocor”, “Kebocoran data Tokopedia”, “Tokopedia diretas”. Kueri tingkat lanjut kemudian dimanfaatkan untuk mendapatkan *twit* balasan dari *thread* yang memiliki banyak interaksi *reply*. Kueri tingkat lanjut yang digunakan: “Tokopedia bocor to:secgron”, “Tokopedia bocor to:tokopedia”. Contoh *twit* dengan banyak interaksi *reply* bisa dilihat pada Gambar 2.

Gambar 2. Contoh *Twit* dengan Banyak Interaksi *Reply*.

Pengambilan data dilakukan secara otomatis menggunakan *crawler* yang merupakan *fork package* Python TweetScrapper oleh jonbakerfish, yang kemudian dikembangkan lebih lanjut oleh penulis. Sumber kode *crawler* yang digunakan pada penelitian ini dapat diakses di https://github.com/nadhifikbarw/ep-scrapper (Wibowo, 2020).

Untuk menjalankan *crawler* digunakan perintah sebagai berikut yang akan diulang untuk semua kueri yang telah ditetapkan.

scrapy crawl TweetScrapper -a query="Tokopedia bocor"

Data masing-masing *twit* yang diambil oleh *scrapper* disimpan pada file *plaintext* dengan format *JSON* menggunakan nama sesuai dengan *conversation_id* dari *twit* tersebut (tanpa ekstensi file), sehingga tidak ada data duplikat yang tersimpan. Seluruh data *twit* tersimpan pada folder./Data/tweets.

Setelah *scrapping* selesai dilakukan untuk setiap kueri, seluruh data diagregasi menjadi satu file CSV untuk memudahkan pemrosesan lebih lanjut. Data yang digunakan pada penelitian ini berjumlah 1060 data *twit*. Masing-masing *twit* diberi label positif, negatif, netral, atau tidak relevan sesuai dengan sentimennya. Jika data tidak relevan maka data *twit* tersebut dikeluarkan dari *dataset* (Maulana et al., 2020). Setelah data yang tidak relevan dikeluarkan data yang tersisa berjumlah 494 *twit*. Rekapitulasi data *twit* yang telah diberi label dapat dilihat pada Tabel 1.

Tabel 1. Jumlah Data *Twit* Teragregasi.

Data *Twit*	Jumlah
Twit sentimen positf	15
Twit sentimen negatif	318
Twit sentimen netral	161
Total	494

Adapun contoh *twit* untuk masing-masing kelas dapat dilihat pada Tabel 2.

Tabel 2. Contoh Data *Twit* Teragregasi.

No	*Twit*	Label
1	@TokopediaCare bagaimana bisa perusahaan sebesar Tokopedia datanya bisa bocor gw jadi ragu belanja di tokped	Negatif
2	Enggak ada salahnya mengucapkan terima kasih dan apresiasi. Untuk isu kebocoran data ini, ya jangan panik namun tetap waspada. @tokopedia big thanks for your service. Panjang umur.	Positif
3	15 Juta Data Pengguna Tokopedia Diinformasikan Bocor, Ini Daftar Email & Password Sebagian Korban https://t.co/fkIFEfgng0	Netral
4	Kementerian Komunikasi dan Informatika (Kemenkominfo) akan segera memanggil Direksi Tokopedia. Pemanggilan ini terkait dugaan kebocoran data pribadi 91 juta akun pengguna layanan ecommerce itu.	Netral
5	@ezash @tokopedia Terima kasih bang Eca atas infonyaa	Tidak Relevan

2.2. Pra-Proses Data

Pada fase ini penulis membuat program Python bernama Data Processor (Wibowo, 2020), yang dapat ditemukan pada *repository* yang sama dengan program *crawler*, untuk melakukan proses ETL (*extraction, transformation, load)*. Pada tahapan *transformation* setiap *twit* dibersihkan dari tagar, *username*, alamat tautan, angka, tanda baca, dan emoji setelah itu dilakukan *case folding*. Proses pembersihan data ini memiliki tujuan untuk menghilangkan *noise* yang terdapat pada data mentah dari proses *scrapping*. Menurut (Tang et al., 2005) proses ini dapat mempengaruhi performa dari model yang dihasilkan terutama pada metrik akurasi.

Keseluruhan *twit* kemudian diformalisasi dari kata yang tidak baku, singkatan. Lalu data dibersihkan kembali dari *stopword* karena kata-kata tersebut tidak memberikan kontribusi pada sentimen. Setelah itu data *twit* ditransformasi *stemming* untuk mendapatkan kata dasar dari setiap kata yang ada di data *twit*.

2.3. Proses Data

Data *twit* dilakukan tokenisasi dalam bentuk *unigram* (satu kata) lalu dilakukan pembobotan dengan metode TF-IDF. *Term Frequency* (TF) mengambarkan banyaknya kemunculan suatu kata pada suatu dokumen, dalam hal ini suatu *twit*. *Inverse Document Frequency* (IDF) mengambarkan prioritas suatu kata dari keseluruhan dokumen, dalam hal ini keseluruhan *twit* (Beel et al., 2017). Selanjutnya seluruh *twit* yang telah dilakukan tokenisasi tersebut dibagi dua menjadi data latih dan data uji dengan perbandingan 80% data latih dan 20% data uji.

2.4. Membangun Model

Menyeimbangkan jumlah ketiga kelas dengan menggunakan teknik SMOTE pada data latih. Metode SMOTE melakukan *oversampling* pada kelas minoritas, dengan membuat data sintetis dari kelas minoritas sehingga jumlahnya sama dengan jumlah data kelas mayoritas. Penelitian sebelumnya menunjukan bahwa teknik SMOTE berhasil meningkatkan performa akurasi dari model (Chawla et al., 2002).

Pembentukan model dilakukan dengan memanfaatkan 3 algoritma *classifier*:

2.4.1. Random Forest

Random Forest adalah algoritma non-parametrik. Algoritma ini merupakan bentuk dari metode *ensemble*, yaitu metode yang mengabungkan (*voting*) hasil dari beberapa model yang lebih sederhana. Dengan banyaknya model yang digabungkan maka hasil klasifikasi dapat lebih baik (VanderPlas, 2016).

2.4.2. Logistic Regression

Logistic Regression merupakan metode statisik yang mirip dengan *Linear Regression* karena metode ini menemukan persamaan bersifat logistik yang memprediksi hasil untuk variabel biner (Y) dari satu atau lebih variabel respon (X). Namun, variabel respon (X) dapat bersifat kategorikal atau kontinu (Hoffman, 2019).

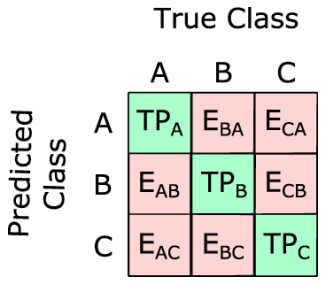
2.4.3. Support Vector Machine

Support Vector Machine merupakan mesin pembelajaran universal yang bisa diterapkan pada regresi maupun pengenalan pola (*pattern recognition*). SVM menggunakan perangkat yang disebut pemetaan kernel (*kernel mapping*) untuk memetakan data dalam ruang input ke ruang fitur berdimensi tinggi di mana masalah menjadi dapat dipisahkan secara linier (Zhang et al., 2004).

2.5. Evaluasi Model

Mengukur performa klasifikasi dari model-model yang dibangun dilakukan dengan memanfaatkan metrik-metrik pengukuran performa yang diturunkan dari pemetaan *confusion matrix.* Berdasarkan (Tharwat, 2020) performa klasifikasi suatu model dapat diwakili oleh nilai skalar seperti metrik-metrik turunan *confusion matrix* layaknya *precision*, *recall*, dan *f1-score*.

Confusion matrix adalah tabel *cross-tabulation* yang memetakan ukuran seberapa baik prediksi model klasifisikasi dibandingkan dengan hasil prediksi model (Lanham & Bedinelli, 2015). Model *confusion matrix* dengan banyak kelas dapat dilihat pada Gambar 3.



Gambar 3. *Confusion Matrix* Banyak Kelas (Tharwat, 2020).

Sumbu diagonal yang berwarna hijau merepresentasikan jumlah prediksi benar sedangkan sel berwarna pink mengindikasikan jumlah prediksi salah yang dihasilkan oleh model. Ketika sampel positif diklasifikasikan dengan kelas positif maka prediksi tersebut *true positive* (TP). Ketika sampel positif diklasifikasikan dengan kelas negatif maka prediksi tersebut *false negative* (FN) atau disebut *Type II error*. Apabila sampel negatif diklasifikasikan dengan kelas positif maka prediksi tersebut *false positive* (FP) yang merupakan *false alarm* atau *Type I error*. Ketika sampel negatif diklasifikasikan dengan kelas negatif maka prediksi tersebut *true negative* (TN). *Confusion matrix* ini digunakan untuk menghitung berbagai macam metrik performa model.

Dalam melakukan perhitungan performa model klasifikasi multi kelas terdapat dua metode dalam melakukan perhitungan metrik, dengan menghitung rata-rata dari metrik yang sama yang dihitung untuk seluruh *classifier* atau yang disebut *macro-average*. Metode kedua dengan menentukan nilai TP, FN, dan TN kumulatif dan kemudian menghitung ukuran kinerja, metode ini disebut dengan metode *micro-averaging.* Metode *macro-averaging* memperlakukan semua kelas secara setara sementara *micro-averaging* lebih menguntungkan kelas yang memiliki sampel yang lebih besar (Sokolova & Lapalme, 2009). Pada studi ini digunakan metode *macro-averaging* untuk mengantisipasi ketidakseimbangan data pada tiap kelas sehingga sehingga nilai metrik akan merefleksikanan performa model ketika memiliki performa buruk dalam klasifkasi suatu kelas.

Sensitivitas, *true positive rate* (TPR), *hit rate*, atau *recall*, adalah metrik mewakili sampel yang diklasifikasikan dengan benar positif dari seluruh jumlah total sampel positif (Tharwat, 2020), dan metrik ini memiliki formula sebagai berikut:

$$Recall_M = \frac{\sum_{i=1}^{i} \frac{TP_i}{(TP_i + FN_i)}}{i} \quad (1)$$

Metrik komplemen lain yang dapat digunakan adalah presisi atau yang juga disebut Nilai Prediksi Positif (PPV) mewakili proporsi sampel positif yang diklasifikasikan dengan benar ke jumlah total sampel yang diprediksi positif seperti yang dihitung menggunakan formula sebagai berikut.

$$Precison_M = \frac{\sum_{i=1}^{i} \frac{TP_i}{(TP_i + FP_i)}}{i} \quad (2)$$

F-measure atay juga disebut *f1-score*, merupakan nilai yang menunjukkan rata-rata harmonik presisi dan *recall* yang dihitung menggunakan persamaaan berikut

$$F1\ Score_M = \frac{2 \times Recall_M \times\ Precison_M}{Recall_M + Precison_M} \quad (3)$$

Nilai *f-measure* berkisar dari nol hingga satu, dan nilai *f-measure* yang tinggi menunjukkan kinerja klasifikasi yang tinggi. Metrik ini sensitif terhadap perubahan dalam distribusi data (Tharwat, 2020).

3. HASIL DAN PEMBAHASAN

Dari 494 data yang relevan terdiri dari 318 data *twit* bersentimen negatif, 161 data *twit* bersentimen netral, dan 15 data *twit* bersentimen positif. Tabel 3 merupakan contoh data setelah dibersihkan dari tanda baca, emoji, dan angka.

Tabel 3. Contoh *Twit* Setelah Pra-Proses.

No	*Twit*	Label
1	bagaimana bisa perusahaan sebesar tokopedia datanya bisa bocor gw jadi ragu belanja di tokped	Negatif
2	enggak ada salahnya mengucapkan terima kasih dan apresiasi untuk isu kebocoran data ini ya jangan panik namun tetap waspada big thanks for your service panjang umur	Positif
3	juta data pengguna tokopedia diinformasikan bocor ini daftar email password sebagian korban	Netral
4	kementerian komunikasi dan informatika kemenkominfo akan segera memanggil direksi tokopedia pemanggilan ini terkait dugaan kebocoran data pribadi juta akun pengguna layanan ecommerce itu	Netral

Setelah itu *twit* dilakukan formalisasi dengan maksudkan untuk mendapatkan kata yang formal dari suatu akronim dan kata yang tidak baku. Misal, ‘yg’ diubah menjadi ‘yang’, ‘abis’ menjadi ‘habis’, dan khusus kasus ini ‘tokped’ menjadi ‘tokopedia’. *Twit* yang telah melalui proses formalisasi dapat dilihat pada Tabel 4.

Tabel 4. Contoh *Twit* Setelah Proses Formalisasi.

No	*Twit*	Label
1	bagaimana bisa perusahaan sebesar tokopedia datanya bisa bocor saya jadi ragu belanja di tokopedia	Negatif
2	tidak ada salahnya mengucapkan terima kasih dan apresiasi untuk isu kebocoran data ini iya jangan panik namun tetap waspada big terima kasih for your service panjang umur	Positif
3	juta data pengguna tokopedia diinformasikan bocor ini daftar email password sebagian korban	Netral
4	kementerian komunikasi dan informatika kemenkominfo akan segera memanggil direksi tokopedia pemanggilan ini terkait dugaan kebocoran data pribadi juta akun pengguna layanan ecommerce itu	Netral

Setelah data *twit* dilakukan *stemming* untuk mendapatkan kata dasar dari tiap kata. Misal, kata ‘mencuri’ dan ‘dicuri’ akan menjadi bentuk dasarnya yaitu ‘curi’. *Stemming* dilakukan untuk memperkecil ukuran data karena setiap kata diperoses pada bentuk dasarnya tanpa mengurangi sentimen yang terkandung pada data tersebut. Contoh *Twit* setelah *stemming* dapat dilihat pada Tabel 5.

Tabel 5. Contoh *Twit* Setelah Proses *Stemming*.

No	*Twit*	Label
1	bagaimana bisa usaha besar tokopedia data bisa bocor saya jadi ragu belanja di tokopedia	Negatif
2	tidak ada salah ucap terima kasih dan apresiasi untuk isu bocor data ini iya jangan panik namun tetap waspada big terima kasih for your service panjang umur	Positif
3	juta data guna tokopedia informasi bocor ini daftar email password bagi korban	Netral
4	menteri komunikasi dan informatika kemenkominfo akan segera panggil direksi tokopedia panggil ini kait duga bocor data pribadi juta akun guna layan ecommerce itu	Netral

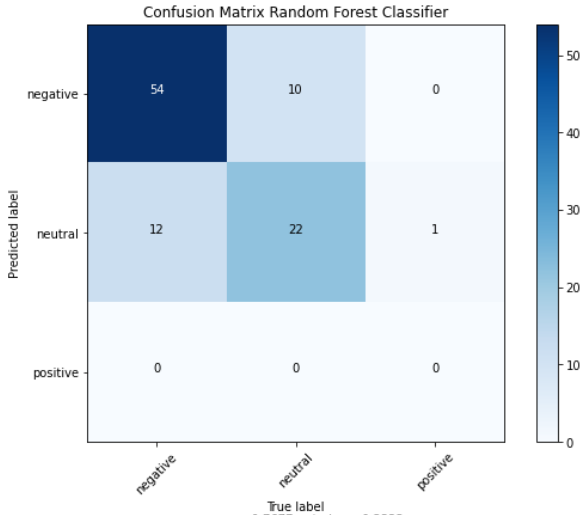
Setelah itu semua *stopword* seperti ”dan”, “itu”, “yang”, dan yang lainnya dihapus dari setiap *twit*. Hal tersebut dilakukan untuk menghilangkan kata yang tidak bermakna bagi sentimen suatu kalimat (*twit*) sehingga dapat memperkecil ukuran data *twit* dan dapat meningkatkan akurasi (Silva & Ribeiro, 2003). Tabel 6 merupakan contoh *Twit* setelah menghilangkan *stopword*.

Tabel 6. Contoh *Twit* Setelah Menghilangkan *Stopword.*

No	*Twit*	Label
1	usaha tokopedia data bocor ragu belanja tokopedia	Negatif
2	salah terima kasih apresiasi isu bocor data iya panik waspada big terima kasih for your service umur	Positif
3	juta data tokopedia informasi bocor daftar email password korban	Netral
4	menteri komunikasi informatika kemenkominfo panggil direksi tokopedia panggil kait duga bocor data pribadi juta akun layan ecommerce	Netral

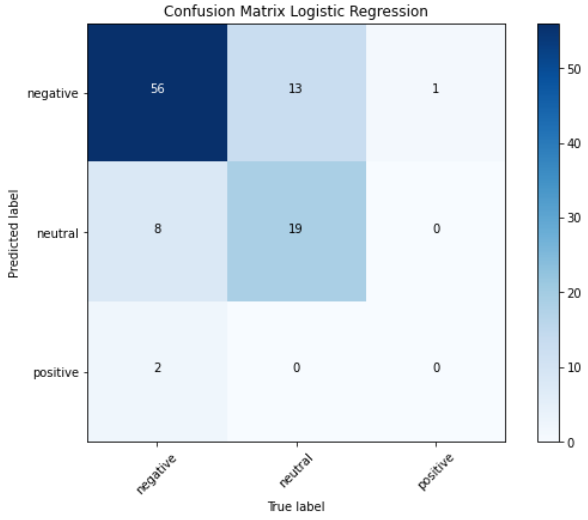
Setelah itu seluruh data *twit* dilakukan transformasi *unigram* dengan pembobotan TF-IDF. Tokenisasi *unigram* merubah *twit* yang mulanya “menteri komunikasi informatika kemenkominfo panggil direksi tokopedia panggil kait duga bocor data pribadi juta akun layan ecommerce” menjadi kumpulan kata seperti “mentri” “komunikasi” “informatika” “kemenkominfo” “panggil” “direksi” “tokopedia” “panggil” “kait” “duga” “bocor” “data” “pribadi” “juta” “akun” “layanan” “ecommerce”. Setelah itu kumpulan data tersebut diberi bobot menggunakan perhitungan TF-IDF. Sehingga diperoleh sebanyak 1329 kata yang menjadi kolom.

Setelah itu data dibagi menjadi data latih dan data uji, data uji terdiri dari *twit* negatif 66.66%, *twit* positif 1.01%, dan *twit* netral 32.32%.



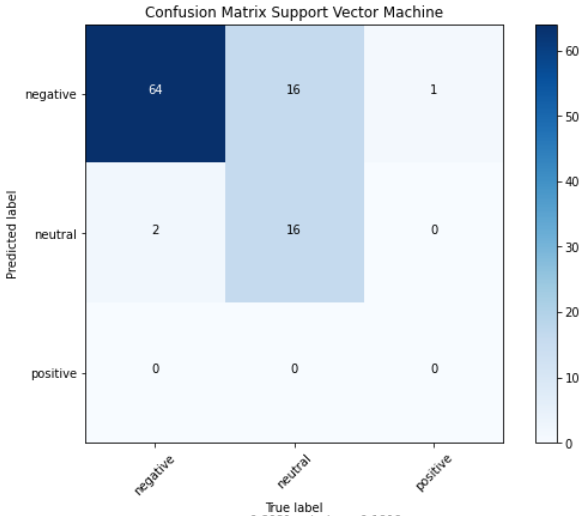
Gambar 4. *Confusion Matrix* Model *Random Forest*.

Berdasarkan Gambar 4 *Random Forest* diketahui bahwa prediksi *twit* negatif dan aktualnya negatif sebanyak 54, prediksi *twit* negatif namun aktualnya netral sebanyak 10, serta prediksi *twit* negatif namun aktualnya positif tidak ada. Selanjutnya, prediksi *twit* netral namun aktualnya negatif sebanyak 12, prediksi *twit* netral dan aktualnya netral sebanyak 22, serta prediksi *twit* netral yang aktualnya positif sebanyak 1. Sementara itu, *confusion matrix* jenis ini tidak bisa memprediksi *twit* bersentimen positif.



Gambar 5. *Confusion Matrix* Model *Logistic Regression*.

Berdasarkan Gambar 5 prediksi *twit* negatif dan aktualnya negatif dari Logistic Regression sebanyak 56, prediksi *twit* negatif namun aktualnya netral sebanyak 13, serta prediksi *twit* negatif namun aktualnya positif hanya ada 1. Kemudian, prediksi *twit* netral namun aktualnya negatif sebanyak 8, prediksi *twit* netral dan aktualnya netral sebanyak 19, serta tidak ada prediksi *twit* netral namun aktualnya positif. Sedangkan untuk prediksi *twit* positif namun aktualnya negatif sebanyak 2, dan tidak terdapat prediksi *twit* positif yang aktualnya netral maupun positif.



Gambar 6. *Confusion Matrix* Model *Support Vector Machine*.

Berdasarkan Gambar 6 hasil prediksi untuk *confusion matrix* ini yaitu, prediksi *twit* negatif dan aktualnya negatif sebanyak 64, prediksi *twit* negatif namun aktualnya netral sebanyak 16, serta prediksi *twit* negatif namun aktualnya positif hanya ada 1. Selanjutnya, prediksi *twit* netral namun aktualnya negatif hanya ada 2, prediksi *twit* netral dan aktualnya netral sebanyak 16, serta tidak terdapat predikti *twit* netral yang aktualnya positif. Berdasarkan gambar, *Confusion matrix* ini tidak mampu membuat prediksi *twit* bersentimen positif.

Setelah diamati dari ketiga kelas yang ada dan dari ketiga *Confusion Matrix* yang telah dibuat tidak ada satupun yang dapat memprediksi *twit* bersentimen positif dengan benar. Ketepatan klasifikasi dipengaruhi olah beberapa faktor, diantaranya jumlah teks atau *term* yang diidentifikasi, jumlah data latihan yang digunakan, fitur klasifikasi, algoritma yang digunakan, dan kemiripan kata yang ada pada saat proses klasifikasi (Faradhillah et al., 2016).

Tabel 7 memberi informasi mengenai *precision, recall*, dan *f1-score* untuk setiap model.

Tabel 7. Hasil Prediksi Model.

Model	Macro Precision	Macro Recall	Macro F1
Random Forest	0.500316	0.501578	0.500829
Logistic Regression	0.501235	0.480745	0.489199
Support Vector Machine	0.559671	0.489899	0.503583

4. KESIMPULAN

Berdasarkan pengamatan yang telah dilakukan terhadap tiga *classifier* berbeda untuk membuat model analisa sentimen *twit* tentang insiden kebocoran data Tokopedia, dapat disimpulkan bahwa *Support Vector Machine* merupakan *classifier* dengan performa terbaik karena dari total 494 *twit* yang dianalisa, *classifier* ini memberikan *f1-score* tertinggi sebesar 0.503583.

Penulis menyarankan penelitian selanjutnya untuk membuat daftar *stopword* yang spesifik untuk suatu *dataset*, karena menggunakan *stopword* yang umum dapat berdampak negatif pada performa model yang diamati (Silva & Ribeiro, 2003). Selain itu penulis menyarankan penelitian selanjutnya untuk mengamati tolok ukur yang lain untuk menilai performa model, seperti AUC pada grafik ROC.

UCAPAN TERIMA KASIH

Dalam pembuatan paper ini, penulis mendapatkan banyak bantuan dari berbagai pihak baik secara langsung maupun tidak langsung sehingga paper ini bisa diselesaikan oleh penulis. Oleh

karena itu, penulis ingin mengucapkan terima kasih kepada semua pihak yang terlibat, diantaranya:
1) Orang Tua yang selalu memberi dukungan moril dan materiil.
2) Dr. Mudjahidin, S.T, M.T. selaku Kepala Departemen Sistem Informasi.
3) Nur Aini Rakhmawati S.Kom., M.Sc.Eng., Ph.D. selaku Dosen Pembimbing dan Pengampu mata kuliah Etika Profesi.

DAFTAR PUSTAKA

Beel, J., Langer, S., & Gipp, B. (2017). TF-IDuF: A Novel Term-Weighting Sheme for User Modeling based on Users' Personal Document Collections. *Proceedings of the iConference 2017*, 1–7.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

CNN Indonesia. (2020). *Deretan Peristiwa Kebocoran Data Warga RI Sejak Awal 2020*. CNN Indonesia. https://www.cnnindonesia.com/teknologi/20200623160834-185-516532/deretan-peristiwa-kebocoran-data-warga-ri-sejak-awal-2020

Deviyanto, A., & Wahyudi, M. D. R. (2018). PENERAPAN ANALISIS SENTIMEN PADA PENGGUNA TWITTER MENGGUNAKAN METODE K-NEAREST NEIGHBOR. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *3*(1), 1. https://doi.org/10.14421/jiska.2018.31-01

Faradhillah, N. Y. A., Kusumawardani, R. P., Hafidz, I., Informasi, J. S., & Informasi, F. T. (2016). Eksperimen Sistem Klasifikasi Analisa Sentimen Twitter Pada Akun Resmi Pemerintah Kota Surabaya Berbasis Pembelajaran Mesin. *Seminar Nasional Sistem Informasi Indonesia*, 15–24.

Hasan, A., Moin, S., Karim, A., & Shamshirband, S. (2018). Machine Learning-Based Sentiment Analysis for Twitter Accounts. *Mathematical and Computational Applications*, *23*(1), 11. https://doi.org/10.3390/mca23010011

Hoffman, J. I. E. (2019). Basic Biostatistics for Medical and Biomedical Practitioners. In *Biostatistics for Medical and Biomedical Practitioners*. Elsevier. https://doi.org/10.1016/C2018-0-02190-8

Lanham, M., & Bedinelli, R. (2015). *Evaluating Stochastic Cost-Benefit Classification Measures for A Retailer's Assortment Mix Decision*.

Librianty, A. (2016, Maret). *Data Jadi Incaran Utama Penjahat Cyber*. Liputan6. https://www.liputan6.com/tekno/read/2466293/data-jadi-incaran-utama-penjahat-cyber

Maulana, T., Rakhmawati, N., Wibowo, N., & Muhammad, H. (2020). *Data Set Sentimen Twit Terhadap Insiden Kebocoran Data Tokopedia* (1.0). Zenodo. https://doi.org/10.5281/ZENODO.4430588

Silva, C., & Ribeiro, B. (2003). The importance of stop word removal on recall values in text categorization. *Proceedings of the International Joint Conference on Neural Networks, 2003.*, *3*, 1661–1666. https://doi.org/10.1109/IJCNN.2003.1223656

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

Tang, J., Li, H., Cao, Y., & Tang, Z. (2005). Email data cleaning. *Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining - KDD '05*, 489. https://doi.org/10.1145/1081870.1081926

Tharwat, A. (2020). Classification assessment methods. *Applied Computing and Informatics*, *17*(1), 168–192. https://doi.org/10.1016/j.aci.2018.08.003

VanderPlas, J. (2016). Python Data Science Handbook: Essential Tools for Working with Data. In *O'Reilly* (1 ed.). O'Reilly Media.

Vardiansyah, D. (2008). *Filsafat Ilmu Komunikasi Suatu Pengantar*. Indeks.

Wibowo, N. (2020). *Program Scrapper Twit Tanpa API dan Pemroses Data* (1.0). Zenodo. https://doi.org/10.5281/zenodo.4231819

Zhang, L., Zhou, W., & Jiao, L. (2004). Wavelet Support Vector Machine. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, *34*(1), 34–39. https://doi.org/10.1109/TSMCB.2003.811113