BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Imelda (2015) dalam penelitian tentang penerapan metode *Support Vector Machine* pada klasifikasi *tweet* agar tidak bercampur antara iklan dan tidak iklan menggunakan kernel RBF (Radial Basis Function) dengan parameter nilai C dan γ. Hasil dari penelitian ini setelah dilakukan pemilihan fitur menghasilkan akurasi terbaik sebesar 99.12% sehingga membantu pengguna Twitter untuk melakukan filter terhadap *tweet* iklan yang terdapat pada akun Twitter mereka.

Rizky (2016) juga melakukan penelitian tentang analisis sentimen terhadap tokoh-tokoh publik Indonesia menggunakan Twitter Streaming API dan metode *Support Vector Machine*. Penelitian ini memanfaatkan pustaka libSVM sebagai salah satu *machine learning* untuk *text classification*, algoritma porter pada proses *stemming* dan metode *Term Frequency* untuk pembobotan. Penelitian dari data 1.400 tweet dan 200 data uji menghasilkan akurasi sebesar 79,5%. Data yang diterima ditampilkan dan divisualisasikan berupa *pie chart* berupa persentase hasil klasifikasi yaitu positif, negatif, netral.

Irene (2017) melakukan penelitian klasifikasi sentimen terhadap *review* film menggunakan metode *Support Vector Machine*. Pengujian dilakukan dengan membandingkan akurasi kernel linear dan non-linear pada SVM yaitu kernel *Radial Basis Function* (RBF) dan *polynomial*. Hasil penelitian membuktikan bahwa semakin banyak data yang digunakan pada proses *training*, semakin tinggi nilai F1-Score yang dihasilkan oleh sistem dalam melakukan klasifikasi. Nilai F1-Score

yang paling baik pada pembagian data 90% data training dan 10% data *testing* dengan hasil 85.6%. Pada pengujian menggunakan *linear separable* dan *non-linear separable* didapatkan nilai F1-Score yang baik sebesar 84.9% pada kasus *linear separable*.

Valonia (2017) melakukan penelitian penerapan sentimen analisis pada hasil evaluasi dosen dengan metode *Support Vector Machine*. Penelitian ini melakukan sentimen analisis terhadap hasil evaluasi dosen FTI UKDW semester gasal dengan 3 kelas sentimen. Akurasi tertinggi yang dihasilkan sistem yaitu sebesar 67,83% dengan metode K-fold cross validation.

Pravina (2019) melakukan penelitian analisis sentimen terhadap opini maskapai penerbangan pada dokumen Twitter menggunakan algoritma *Support Vector Machine*. Penelitian ini melakukan klasifikasi dengan fitur *Lexicon Based* menghasilkan kelas positif dan negatif dari *tweet* berbahasa Inggris. Penelitian ini dengan parameter C bernilai 10 dan learning rate bernilai 0,03 serta *Lexicon Based Features* dengan iterasi sebanyak 50 kali memberikan hasil *accuracy* sebesar 40%, *precision* 40%, 100% *recall*, dan f-measure sebesar 57,14%.

Penelitian yang akan dilakukan adalah mengimplementasikan metode *Support Vector Machine* menggunakan *Kernel Radial Basis Function* dan pembobotan TF-IDF untuk pengelompokan kategori positif, negatif, netral tentang opini publik terhadap pemerintah terkait kasus virus corona berdasarkan beberapa referensi penelitian yang dapat dilihat pada Tabel 2.1.

Tabel 2.1 Perbandingan Penelitian

Nama Peneliti	Metode dan Teknologi	Objek	Hasil Penelitian
Imelda A.Muis dan Muhammad Affandes, M.T (2015)	*Support Vector Machine* dengan *Kernel Radial Basis Function*	Iklan pada Twitter	Klasifikasi Tweet Iklan dan Tidak Iklan dengan Nilai Akurasi 99.12%.
Rizky Maulana (2016)	*Support Vector Machine* dengan LibSVM	Opini Pengguna Twitter terhadap Tokoh Publik	Persentase Tweet Positif, Netral, Negatif dengan Akurasi 79,5%.
Irene Mathilda Yulietha, dkk (2017)	*Support Vector Machine* dengan Perbandingan *Kernel Linear* dan *Non Linear*	Review Film	Klasifikasi Review Film dengan F1-Score 84.9%
Valonia Inge Santoso, dkk (2017)	*Support Vector Machine* dengan Penerapan K-fold cross validation	Hasil Evaluasi Dosen FTI UKDW	Klasifikasi Sentimen Positif, Negatif, Netral dengan Akurasi 67,83%.
Arsya Monica Pravina, dkk (2019)	*Support Vector Machine* dengan Fitur *Lexicon Based*	Opini terhadap Maskapai Penerbangan	Klasifikasi Tweet Positif dan Negatif dengan akurasi 40%
Helda Ludya Safitri (2020)	Analisis Sentimen Tindakan Pemerintah Indonesia Pada Kasus Covid-19 Menggunakan Metode *Support Vector Machine*	Opini Publik terhadap Kasus Covid-19 pada Tweet	Persentase Sentimen Positif, Netral, Negatif pada Tweet Opini Covid-19 dengan akurasi 77%

2.2 Dasar Teori

2.2.1 Twitter

Twitter adalah sebuah layanan jejaring sosial (media sosial) dan mikroblog yang memungkinkan penggunanya berkirim dan membaca pesan yang tidak lebih dari 280 karakter yang disebut sebagai *tweet.* Twitter sebagai salah satu media komunikasi memungkinkan para pengguna untuk berinteraksi. Pengguna dapat mengirimkan pesan kepada pengguna lain baik secara personal ataupun terbuka melalui twitter dengan adanya akses internet.

Sebagai salah satu layanan terbesar, pengguna Twitter telah berkembang pesat dan menarik perhatian dari banyak perusahaan terhadap kebiasaan pelanggan. Twitter digunakan oleh organisasi berita untuk menerima informasi tentang bahaya dan bencana alam. Beberapa bisnis dan organisasi menggunakannya untuk menyampaikan informasi ke *stakeholder*.

2.2.2 Analisis Sentimen

Analisis sentimen merupakan bidang studi yang menganalisis pendapat, sentimen, penilaian, evaluasi, sikap, dan emosi seseorang terkait suatu topik, layanan, produk, individu, organisasi, atau kegiatan tertentu (Liu, 2012). Analisis sentimen digunakan untuk menganalisis *tweet* di Twitter kemudian diterjemahkan menjadi sesuatu yang lebih bermakna, salah satunya dalam bentuk statistik sederhana mengenai persentase sentimen positif, negatif, netral terhadap opini publik.

Analisis sentimen terdiri dari tiga level analisis yaitu:

1. Level Dokumen

Level dokumen menganalisis satu dokumen penuh dan mengklasifikasikan dokumen tersebut ke dalam sentimen positif atau negatif. Level analisis ini berasumsi bahwa keseluruhan dokumen hanya berisi opini tentang satu entitas saja. Level analisis ini tidak cocok diterapkan pada dokumen yang membandingkan lebih dari satu entitas (Liu, 2012).

2. Level Kalimat

Level kalimat menganalisis satu kalimat dan menentukan setiap kalimat bernilai positif, netral, atau negatif. Sentimen netral berarti kalimat tersebut bukan opini (Liu, 2012).

3. Level Entitas dan Aspek

Level aspek tidak melakukan analisis pada konstruksi bahasa (dokumen, paragraf, kalimat, klausa, atau frasa) melainkan langsung pada opini itu sendiri. Hal ini didasari bahwa opini terdiri dari sentimen (positif atau negatif) dan target dari opini tersebut. Tujuan level analisis ini adalah untuk menemukan sentimen entitas pada tiap aspek yang dibahas (Liu, 2012).

2.2.3 Preprocessing

Tahap *preprocessing* atau praproses data merupakan proses untuk mempersiapkan data mentah sebelum dilakukan proses lain. Pada umumnya, praproses data dilakukan dengan cara mengeliminasi data yang tidak sesuai atau mengubah data menjadi bentuk yang lebih mudah diproses oleh sistem. Praproses sangat penting dalam melakukan analisis sentimen, terutama untuk media sosial yang sebagian besar berisi kata - kata atau kalimat yang tidak formal dan tidak terstruktur serta memiliki *noise* yang besar (A Clark, 2003).

Tahap *preprocessing* memiliki beberapa tahap proses sebagai berikut:

- *Text Cleaning* bertujuan menghapus karakter seperti URL, *hashtag*, *link*.
- *Case Folding* bertujuan mengubah semua huruf pada dokumen menjadi huruf kecil.
- *Tokenizing* bertujuan untuk memotong string input berdasarkan setiap kata yang menyusunnya.
- *Stopwords Removal* bertujuan untuk mengambil kata-kata yang sering muncul dan tidak memiliki makna pada teks dari hasil *tokenizing*.

2.2.4 Term Frequency-Inverse Document Frequency

Pembobotan *Term Frequency-Inverse Document Frequency* (TF-IDF) adalah salah satu proses dari teknik ekstraksi fitur dengan memberikan nilai pada masing-masing kata (*term*) yang ada pada *tweet*, dimana *term* adalah kata hasil dari proses *preprocessing*, tf menyatakan jumlah berapa banyak keberadaan suatu *term* (t) dalam satu dokumen (d), df merupakan perhitungan banyaknya dokumen dimana suatu *term* (t) muncul dan idf berfungsi mengurangi bobot suatu *term* jika kemunculannya banyak tersebar diseluruh koleksi dokumen.

Pemberian skor pada TF-IDF berdasarkan frekuensi munculnya kata dalam dokumen. Proses awal dilakukan perhitungan kata (term) pada tiap dokumen untuk mendapatkan frekuensi term (tf), kemudian dilakukan perhitungan document frequency (df). Perhitungan inverse document frequency (idf) dilakukan dengan persamaan 2.1 sebagai berikut :

$$idf\,(t) = Log\,(\frac{D}{df}) + 1 \quad .....(2.1)$$

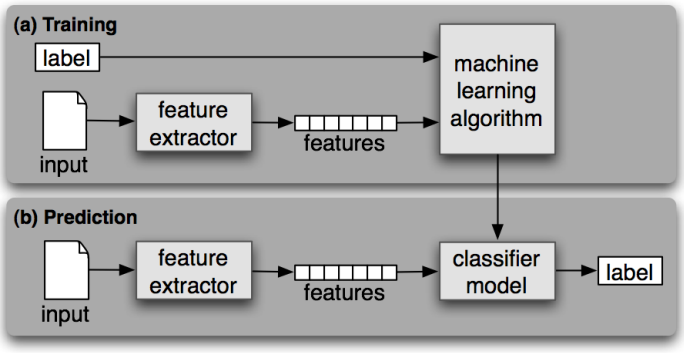
Kemudian perhitungan TF-IDF untuk mendapatkan bobot *term* dengan persamaan 2.2 sebagai berikut :

$$tf\text{-}idf\ (t) = tf\ (t,d) * idf\ (t) \ldots\ldots (2.2)$$

Hasil dari nilai bobot tersebut dapat digunakan dalam pembentukan vektor untuk melakukan klasifikasi.

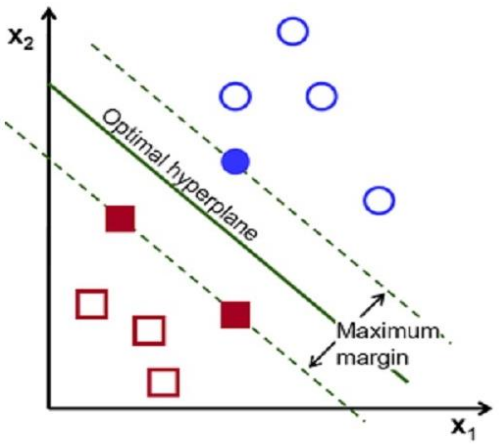
2.2.5 Support Vector Machine

Support Vector Machine (SVM) merupakan salah satu metode klasifikasi *supervised learning* yang memprediksi kelas berdasarkan model atau pola dari hasil proses *training*.



Gambar 2.1 *Supervized Machine Learning*

Klasifikasi dilakukan dengan mencari *hyperplane* atau garis pembatas yang memisahkan antara suatu kelas dengan kelas lain. Dalam kasus ini garis tersebut berperan memisahkan *tweet* bersentimen positif (berlabel +1) dengan tweet bersentimen selain positif (berlabel -1). SVM melakukan pencarian nilai *hyperlane* yang paling maksimal dengan menggunakan *support vector* dan nilai margin (J. Han, 2006). *Hyperplane* pemisah terbaik antara kedua kelas yang diilustrasikan pada Gambar 2.2 ditemukan dengan mengukur margin *hyperplane* tersebut. Margin adalah jarak antara *hyperplane* dengan *pattern* terdekat dari masing-masing kelas. *Pattern* yang paling dekat ini disebut sebagai *support vector*.



Gambar 2.2 Optimal *Hyperplane* SVM

Pada prinsipnya SVM bekerja secara linear dengan klasifikasi yang diasumsikan terdapat dua kelas -1 dan +1 yang terpisah sempurna oleh *hyperplane*, didefinisikan pada fungsi persamaan 2.3 :

$$f(x)= w^T x + b.....(2.3)$$

X yang termasuk kelas -1 dapat dirumuskan sebagai *pattern* yang memenuhi pertidaksamaan 2.4 :

$$w^T x + b \leq -1 .....(2.4)$$

X yang termasuk kelas +1 dirumuskan sebagai *pattern* yang memenuhi pertidaksamaan 2.5 :

$$w^T x + b \geq +1 .....(2.5)$$

SVM dikembangkan untuk dapat diterapkan pada masalah non-linear dengan menggunakan metode *kernel trick* yang mencari *hyperplane* dengan cara mentransformasi *dataset* ke ruang vektor berdimensi lebih tinggi (feature space), kemudian proses klasifikasi dilakukan pada *feature space* tersebut. *Feature space* dalam prosesnya biasanya memiliki dimensi yang lebih tinggi dari vektor input (input space). Hal ini mengakibatkan komputasi pada *feature space* akan menjadi

sangat besar, karena ada kemungkinan *feature space* akan memiliki jumlah fitur yang tidak terhingga. Penentuan fungsi kernel yang digunakan akan sangat berpengaruh terhadap hasil prediksi.

Sebuah fungsi bisa menjadi fungsi kernel jika memenuhi Teorema Mercer, yang menyatakan bahwa matriks kernel yang dihasilkan harus bersifat semi positive semi definite. Berikut ini adalah beberapa fungsi kernel yang umum digunakan yaitu:

a. Kernel Linier : $K(x_i,x) = x_i^T x$

b. Polynomial : $K(x_i,x) = (\gamma.x_i^T x + r)^p$,$\gamma > 0$

c. Radial Basis Function : $K(x_i,x) = \exp(-\gamma||x_i-x||^2),\gamma>0$

d. Sigmoid Kernel : $K(xi,x) = \tanh(\gamma x_i^T + r)$

Pada penelitian ini akan diterapkan kernel *Radial Basis Function* (RBF) dengan parameter C dan Gamma (γ). Untuk mendapatkan fitur baru berdimensi tinggi, dilakukan kernelisasi menggunakan fungsi persamaan 2.6 sebagai berikut :

$$K(x_i,x) = exp(-\gamma||x_i-x||^2).....(2.6)$$

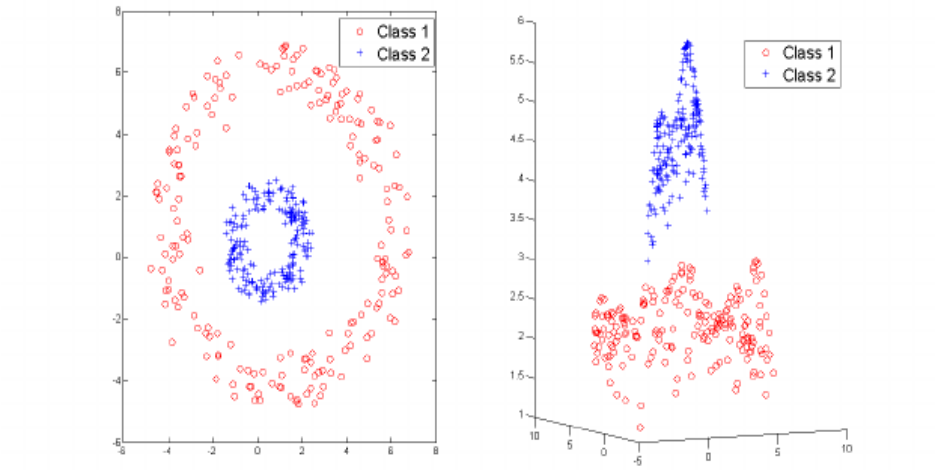
dimana x_i-x adalah data input. Fungsi persamaan 2.6 menghasilkan matriks K yang akan berkorelasi dengan $\alpha_i\alpha_j$ dalam dualitas Lagrange Multiplier (Ld max). Lagrange multiplier digunakan untuk menentukan sejumlah *support vector* dengan menggunakan Quadratic Programming pada persamaan 2.7 :

$$Ld = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i \, . \, x_j .....(2.7)$$

dimana hasil nilai α harus memenuhi :

- Syarat 1 : $\alpha_1 - \alpha_2 - \alpha_3 = 0$
- Syarat 2 : $\alpha_1, \alpha_2, \alpha_3 \geq 0$

Data non linear hasil klasifikasi SVM dengan kernel *Radial Basis Function* (RBF) dalam dimensi yang lebih tinggi dapat diilustrasikan pada Gambar 2.3.



Gambar 2.3 Ilustrasi Data Non Linear dengan Kernel RBF

2.2.6 Scikit-learn

Scikit-learn adalah modul Python yang mengintegrasikan berbagai algoritma pembelajaran *machine learning* untuk masalah berskala menengah yang diawasi dan tidak terawasi. Paket ini berfokus pada "membawa *machine learning* ke non-spesialis" menggunakan *general-purpose high-level language*. Kelebihan dari modul ini ada pada kemudahan penggunaan, kinerja, dokumentasi, dan konsistensi API. Modul ini memiliki ketergantungan minimal dan didistribusikan di bawah lisensi BSD yang disederhanakan, mendorong penggunaannya dalam pengaturan akademik dan komersial (Pedregosa, 2011).

2.2.7 Pengukuran Kinerja Klasifikasi

Pengukuran kinerja klasifikasi pada suatu sistem merupakan hal yang penting. Kinerja sistem klasifikasi menggambarkan seberapa baik sistem

mengklasifikasi data. *Confusion matrix* merupakan salah satu metode yang dapat digunakan untuk mengukur kinerja suatu metode klasifikasi. Pada dasarnya *confusion matrix* mengandung informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan hasil klasifikasi yang seharusnya.

Berdasarkan jumlah keluaran kelasnya, sistem klasifikasi dapat dibagi menjadi 4 (empat) jenis yaitu klasifikasi *binary, multi-class, multi-label* dan *hierarchical* . Pada klasifikasi *binary,* data masukan dikelompokkan ke dalam salah satu dari dua kelas. Jenis klasifikasi ini merupakan bentuk klasifikasi yang paling sederhana dan banyak digunakan. Sementara itu, pada bentuk klasifikasi *multi-class,* data masukan diklasifikasikan menjadi beberapa kelas. Bentuk klasifikasi *multi-label* pada dasarnya sama dengan *multi-class* dimana data dikelompokkan menjadi beberapa kelas, namun pada klasifikasi *multi-label*, data dapat dimasukkan dalam beberapa kelas sekaligus. Bentuk klasifikasi yang terakhir adalah *hierarchical.* Data masukan dikelompokkan menjadi beberapa kelas, namun kelas tersebut dapat dikelompokkan kembali menjadi kelas-kelas yang lebih sederhana secara hirarkis.

Pada pengukuran kinerja menggunakan *confusion matrix,* terdapat 4 istilah sebagai representasi hasil proses klasifikasi. Keempat istilah tersebut adalah *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN). Nilai *True Negative* (TN) merupakan jumlah data negatif yang terdeteksi dengan benar, sedangkan *False Positive* (FP) merupakan data negatif namun terdeteksi sebagai data positif. *True Positive* (TP) merupakan data positif

yang terdeteksi benar. *False Negative* (FN) merupakan kebalikan dari *True Positive*, sehingga data positif, namun terdeteksi sebagai data negatif.

Berdasarkan nilai *True Negative* (TN), *False Positive* (FP), *False Negative* (FN), dan *True Positive* (TP) dapat diperoleh nilai akurasi, presisi, *recall* dan *F1-Score*. Nilai akurasi menggambarkan seberapa akurat sistem dapat mengklasifikasikan data secara benar. Dengan kata lain, nilai akurasi merupakan perbandingan antara data yang terklasifikasi benar dengan keseluruhan data. Presisi menghitung ketepatan antara informasi yang diminta pengguna dengan jawaban sistem. Recall menunjukkan tingkat keberhasilan sistem dalam menemukan informasi. F1-Score merupakan perbandingan rata-rata presisi dan recall. Pada klasifikasi dengan jumlah keluaran kelas yang lebih dari dua (*multi-class*), cara menghitung akurasi, presisi, recall, dan F1-Score dapat dilakukan dengan persamaan berikut :

$$Presisi = \frac{TP_i}{TP_i+FP_i} \times 100\% \text{ .....(2.8)}$$

$$Recall = \frac{TP_i}{TP_i+FN_i} \times 100\% \text{ .....(2.9)}$$

$$F1\text{-}Score = \frac{TP_i}{TP_i+\frac{1}{2}(FP_i+FN_i)} \times 100\% \text{.....(2.10)}$$

$$Akurasi = \frac{TP_i}{Total\ Data} \times 100\% \text{ .....(2.11)}$$

Keterangan :

- TP_i adalah *True Positive*, yaitu jumlah data positif yang terklasifikasi oleh sistem untuk kelas ke-i.

- TN_i adalah *True Negative*, yaitu jumlah data bukan positif yang terklasifikasi oleh sistem untuk kelas ke-i.
- FN_i adalah *False Negative*, yaitu jumlah data negatif namun terklasifikasi salah oleh sistem untuk kelas ke-i.
- FP_i adalah *False Positive*, yaitu jumlah data positif namun terklasifikasi salah oleh sistem untuk kelas ke-i.