BAB 2

TINJAUAN PUSTAKA

2.1 Analisis Sentimen

Analisis sentimen atau opinion mining adalah proses memahami, mengekstrak dan mengolah data tekstual secara otomatis dalam mendapatkan sebuah informasi sentimen yang terdapat dalam suatu kalimat atau opini. Analisis sentimen dilakukan untuk melihat pendapat atau kecenderungan pendapat, sentimen, penilaian, emosi, sikap seseorang terhadap individu, organisasi, peristiwa, apakah cenderung berpandangan atau beropini negatif atau negatif. Besarnya pengaruh dan manfaat dari analisis sentimen ini menyebabkan penelitian dan aplikasi berbasis analisis sentimen berkembang pesat. Bahkan di Amerika terdapat sekitar 20-30 perusahaan yang memfokuskan pada layanan analisis sentimen[7].

Secara umum analisis sentimen merupakan sebuah proses klasifikasi dokumen tekstual ke dalam dua kelas atau lebih, seperti kelas sentimen positif dan kelas sentimen negatif. Analisis sentimen dilakukan untuk melihat pendapat atau opini terhadap sebuah masalah atau dapat juga digunakan untuk identifikasi kecenderungan suatu hal pada lingkup tertentu. Menurut Liu, tujuan dari analisis sentimen adalah untuk melakukan ekstraksi atribut pada sebuah dokumen atau teks berisi opini untuk mengetahui maksud yang ada di dalamnya sehingga opini tersebut dapat dikategorikan positif atau negatif[8].

2.2 Text Mining

Text mining didefinisikan sebagai suatu proses dalam menggali sebuah informasi dimana seseorang berinteraksi dengan sekumpulan dokumen menggunakan tools analisis yang merupakan komponen pada sebuah data mining yang salah satunya adalah kategorisasi. Teks mining dari website yang berisikan sebuah komentar, opini, feedback, kritik dan review merupakan hal yang penting, karena apabila hal ini dikelola dengan baik maka akan dapat memberikan dampak

berupa keuntungan informasi yang bermanfaat untuk membantu individu atau organisasi dalam pengambilan sebuah keputusan. Terdapat kategori yang termasuk dalam teknik text mining salah satunya adalah analisis sentimen, yaitu suatu proses memahami, mengekstrak, dan mengolah data tekstual secara otomatis[9].

2.3 Twitter

Twitter adalah situs web yang dimiliki dan dioperasikan oleh Twitter, Inc., yang menawarkan jaringan sosial berupa microblog. Disebut microblog karena situs ini memungkinkan penggunanya mengirim dan membaca pesan blog seperti pada umumnya namun terbatas hanya sejumlah 140 karakter yang ditampilkan pada halaman profil pengguna. Twitter memiliki karakteristik dan format penulisan yang unik dengan simbol ataupun aturan khusus. Pesan dalam Twitter dikenal dengan sebutan tweet[10].

2.4 Sarkasme

Bahasa sebagai alat komunikasi seseorang untuk menyampaikan aspirasi yang dipikirkan dengan tingkat ekspresif yang dalam menggunakan bahasa, mayoritas pesan yang disampaikan dalam komentar atau tweet cenderung didominasi dengan sentimen negatif yang kerap disampaikan dalam bentuk sarkasme maupun perkataan negatif secara vulgar dan frontal yang terkadang menyebabkan pertengkaran antara netizen dan selebriti. Sarkasme mengandung kepahitan dan celaan yang kasar karena bersifat merendahkan atau mengejek suatu individu atau organisasi[2].

2.5 Klasifikasi

Klasifikasi sebuah aspek penting dari data mining yang merupakan sebuah teknik pemodelan prediktif. Teknik klasifikasi digunakan di berbagai permasalahan dalam berbagai penelitian. Menurut Joshi, klasifikasi dapat digunakan dalam membangun sebuah struktur dari contoh keputusan sebelumnya yang dapat digunakan untuk membuat keputusan di masa yang akan datang. Klasifikasi adalah proses pembagian data menjadi sebuah kelompok yang sifatnya bisa saling

independen atau dependen dan setiap kelompok berperan sebagai sebuah kelas. Terdapat teknik-teknik yang dapat digunakan dalam fungsi klasifikasi dengan beberapa algoritma seperti C4.5, Naïve Bayes (NB), K - Nearest Neighbour (KNN) dan Support Vector Machine (SVM)[11].

2.6 Preprocessing

Sebuah data yang didapat dari hasil pengumpulan data belum bisa langsung diklasifikasikan karena masih terdapat simbol dan kata-kata yang tidak diperlukan. Tahapan yang dilakukan pada preprocessing yaitu : Case Folding, Cleaning, Stopword Removal, Convert Negation, Normalization dan Tokenizing[12].

1. Case Folding

Case Folding adalah proses penyeragaman bentuk huruf menjadi huruf kecil (lowercase). Jika dalam suatu dokumen terdapat beberapa huruf kapital seperti dalam awalan kalimat, nama orang, nama tempat, dll.

2. Cleaning

Pada tahap cleaning akan dilakukan pembersihan kata atau simbol yang tidak diperlukan pada proses hasil klasifikasi sentimen. Dalam data sentimen pasti terdapat banyak atribut-atribut yang tidak diperlukan maka pada tahap ini akan dilakukan penghapusan simbol atau emoticon, penghapusan angka, dan juga penghapusan tanda baca.

3. Stopword Removal

Stopwords removal adalah kata umum yang biasanya muncul dalam jumlah besar dan dianggap tidak memiliki makna, sehingga tidak akan mempengaruhi sentimen. *Stop words* biasanya seperti kata kerja, kata sifat, kata keterangan, dan kata sambung[13].

4. Tokenizing

Tokenizing adalah sebuah proses memisahkan kata dengan spasi. Sebagai contoh karakter whitespace seperti enter, spasi, tabulasi dianggap sebagai pemisah kata. Namun karakter petik (‘), titik (.), titik dua (:), semicolon (:) dapat memiliki peran sebagai pemisah kata.

5. Normalization

Normalization merupakan perbaikan dari kata singkatan menjadi bahasa baku, pada tahap ini menggunakan bantuan kamus. Karena pada sebuah tweet terdapat kata-kata singkatan seperti "Adlh" yang seharusnya "Adalah" ataupun "Gw" yang seharusnya "Saya" dan proses ini dilakukan untuk memperbaiki kesalahan ketik dalam penulisan[14].

6. Convert Negation

Pada tahap ini, setiap tweets yang mengandung kata-kata yang bersifat negasi akan diubah nilai sentimennya. Kata yang bersifat negasi seperti "tidak", "enggak", "ga", "jangan", "nggak", "bukan", "tak", dan "gak". Convert negation dilakukan jika terdapat kata negasi pada sebelum kata yang bernilai positif, maka pada kata tersebut akan diubah nilainya menjadi negatif dan sebaliknya[15]. Tahapan *convert negation* adalah sebagai berikut:

a. Kata yang digunakan adalah hasil dari *normalization*.
b. Setiap kata akan diperiksa dan dibandingkan dengan kumpulan kata-kata yang bersifat negasi.
c. Jika kata yang bersifat negasi terdapat kata yang termasuk sentimen positif, maka sentimen tersebut akan diubah menjadi negatif.
d. Jika setelah kata yang bersifat negasi terdapat kata yang termasuk sentimen negatif, maka sentimen tersebut akan diubah menjadi positif

2.7 Pembobotan TF-IDF

Pembobotan Term Frequency-Inverse Document Frequency (TF-IDF) adalah metode yang digunakan untuk menghitung bobot setiap kata yang telah diekstrak. Setiap kalimat dalam dokumen diekstraksi menggunakan metode *Term frequency-Inverse document frequency* (Tf-Idf) dan dipilih berdasarkan hasil penelitian yang telah dilakukan sebelumnya[16]. Penggunaan metode ini umumnya dilakukan untuk menghitung kata umum yang ada pada information retrieval. Model pembobotan TF-IDF merupakan metode yang mengintegrasikan model term frequency (tf) dan inverse document frequency (idf), dimana term frequency (tf)

merupakan proses untuk menghitung jumlah kemunculan term dalam satu dokumen dan inverse document frequency (idf) digunakan untuk menghitung term yang muncul di berbagai dokumen(komentar) yang dianggap sebagai term umum, yang dinilai tidak penting[17].

Proses awal yang dilakukan dalam pembobotan TF-IDF dilakukan dengan menghitung term frequency ($tf_{i,d}$). Dimana t menunjukkan term dalam dokumen d yang berfungsi untuk menunjukkan kemunculan term t pada dokumen d. Hal ini berpengaruh dalam bobot term yang akan semakin tinggi ketika banyak term yang muncul dalam suatu dokumen. Nilai dari tf akan dihitung bobotnya dengan persamaan 2.1. Rumus tersebut ditunjukkan pada persamaan 2.1.

Metode yang digunakan untuk pembobotan kata pada penelitian ini dapat dirumuskan sebagai berikut :

$$TF * IDF\ (d,t) = TF(d,t) * \log(\frac{D}{df(t)}) \quad (2.1)$$

Banyaknya kata yang muncul pada dokumen, umumnya merupakan nilai term frequency dari kata yang tidak penting. Untuk menghindari pembobotan pada kata tidak penting maka digunakan pembobotan document frequency yang bermaksud untuk menghitung jumlah dokumen yang mengandung term t.

Dari nilai term pada setiap dokumen yang telah ditemukan akan dilakukan proses kebalikan dari pembobotan document frequency. Proses pembobotan ini disebut dengan inverse document frequency, yang menyatakan bahwa frekuensi dari term yang rendah pada banyak dokumen akan memberikan bobot paling tinggi. Perhitungan ini ditunjukkan dengan rumus persamaan 2.2.

$$idft = \frac{D}{dft} \quad (2.2)$$

Keterangan :

idf_t = bobot inverse dari nilai df

D = jumlah dokumen pada kumpulan dokumen

df_t = jumlah dokumen yang mengandung term

Perhitungan pembobotan TF-IDF merupakan perkalian yang dilakukan dari pembobotan term frequency dengan inverse document frequency. Hal ini ditunjukkan pada persamaan 2.3.

$$W_{t,d} = W_t tf_{t,d} \times idf_t \quad (2.3)$$

Keterangan :

$W_t f_{t,d}$ = bobot kata dalam setiap dokumen

$tf_{t,d}$ = jumlah kemunculan kata t pada dokumen d

idf_t = bobot inverse dari nilai df

$W_{t,d}$ = Pembobotan TF-IDF

2.8 Support Vector Machine (SVM)

Support Vector Machine (SVM) merupakan salah satu metode machine learning yang mengubah text menjadi data vector. Vector dalam penelitian ini memiliki dua komponen yaitu dimensi (word id) dan bobot. Konsep SVM dapat dijelaskan secara sederhana sebagai usaha mencari Hyperplane terbaik yang berfungsi sebagai pemisah dua buah class pada input space[18].

Dalam linear Support Vector Machine permisah merupakan fungsi linear. Data latih dinyatakan oleh (xi, yi) dan xi = { x1, x2 ..., xiq} merupakan atribut (fitur) aet untuk data latih kelas ke-i. Untuk yi € {-1,1} menyatakan label kelas[24]. Pendefinisian persamaan suatu hyperplane pemisah yang dituliskan dengan

$$wx_i + b = 0 \quad (2.4)$$

Data xi yang terbagi dalam dua kelas, yang termasuk kelas -1 (sampel negatif) didefinisikan sebagai vektor yang memenuhi pertidaksamaan 2.5 berikut ini.

$$wx_i + b < 0 \; untuk \; y_i = 1 \quad (2.5)$$

Sedangkan yang termasuk kelas +1 (sampel positif) memenuhi pertidaksamaan 2.6 berikut.

$$wx_i + b > 0 \; untuk \; y_i = +1 \quad (2.6)$$

Dimana :

x_i = data input

y_i = label yang diberikan

w = nilai bidang relatif terhadap pusat koordinat

b = posisi bidang relatif terhadap pusat koordinat

Parameter w dan b adalah parameter yang akan dicari nilainya. Bila label data yi = -1, maka pembatas menjadi persamaan 2.7 sebagai berikut :

$$wx_i + b \leq -1 \quad (2.7)$$

Bila label data yi = +1, maka pembatas menjadi persamaan 2.8 berikut.

$$wx_i + b \geq +1 \quad (2.7)$$

Margin terbesar dapat dicari dengan cara memaksimalkan jarak antar bidang pembatas kedua kelas dan titik terdekatnya, yaitu $\frac{2}{|w|}$ Hal ini dirumuskan sebagai permasalahan quadratic programming (QP) problem yaitu mencari titik minimal persamaan (2.8) dengan memperhatikan persamaan (2.9) berikut.

$$\frac{2}{|w|} \; ||w||^2 \quad (2.8)$$

$$y_i(wx_i + b) - 1 \geq 0, (i = 1, \dots, n) \quad (2.9)$$

Permasalahan ini dapat dipecah dengan berbagai teknik komputasi. Lebih mudah diselesaikan dengan mengubah persamaan 2.8 kedalam fungsi lagrange pada persamaan 2.10, dan menyederhanakannya menjadi persamaan 2.11 berikut.

$$L(w, a, b) = \frac{1}{2} \; ||w||^2 - \sum_{i=1}^{n} \quad \alpha_i(y_i\big((w^T x_i + b) - 1\big)) \quad (2.10)$$

$$L(w, a, b) = \frac{1}{2} \; ||w||^2 - \sum_{i=1}^{n} \quad \alpha_i y_i(w^T x_i + b) + \sum_{i=1}^{n} \quad \alpha_i \quad (2.11)$$

Dimana ai adalah lagrange multiplier yang bernilai nol atau positif (ai > 0). Nilai optimal dari persamaan 2.11 dapat dihitung dengan meminimalkan L terhadap w,b dan a. Dapat dilihat pada persamaan 2.12 sampai 2.14 berikut ini.

$$\frac{\partial L}{\partial \alpha} \; w - \sum_{i=1,j=1}^{n} \quad \alpha_i y_i x_i \; = 0 \quad (2.12)$$

$$\frac{\partial L}{\partial \alpha} \; \sum_{i=1,j=1}^{n} \quad \alpha_i y_i = 0 \quad (2.13)$$

$$\frac{\partial L}{\partial \alpha} \; w - \sum_{i=1}^{n} \quad \alpha_i y_i x_i \; - \sum_{i=1}^{n} \quad \alpha_i = 0 \quad (2.14)$$

Maka masalah lagrange untuk klasifikasi dapat dinyatakan pada persamaan 2.13 berikut.

$$Min \; L(w, a, b) = \frac{1}{2}||w||^2 - \sum_{i=1}^{n} \quad a_i y_i(w^T x_i + b) + \sum_{i=1}^{n} a_i \quad (2.13)$$

Untuk mencari nilai xi dan yi dapat dilakukan ketika sudah didapatkan nilai tiap kata (term) dari pembobotan dari masing-masing metode seleksi fitur dan inisialisasi kelas. Hasil dari pembobotan diubah ke dalam bentuk format data SVM

(x), sedangkan data kelas menjadi label data SVM (y). Untuk mendapatkan nilai ai, langkah pertama adalah mengubah setiap tweet menjadi nilai vektor (support vector) $\left(\frac{x}{y}\right)$. Kemudian nilai vektor dari setiap tweet dimasukkan ke persamaan 2.14 kernel trick phi (ϕ) berikut ini.

$$\emptyset \begin{bmatrix} x \\ y \end{bmatrix} = \begin{cases} \sqrt{x_n^2 + y_n^2} > maka \begin{bmatrix} \sqrt{x_n^2 + y_n^2} - x + |x - y| \\ \sqrt{x_n^2 + y_n^2} - x + |x - y| \end{bmatrix} \\ \sqrt{x_n^2 + y_n^2} \leq 2\ maka \begin{bmatrix} x \\ y \end{bmatrix} \end{cases} \quad (2.14)$$

Nilai x didapatkan dari persamaan 2.15 kernel linear untuk x dan y berikut.

$$\sum_{i=1,j=1}^{n} \quad x_i^T x_j (i,j = 1, \dots, n) \quad (2.15)$$

Nilai y didapatkan dari persamaan 2.16 kernel linear untuk x dan y berikut.

$$\sum_{i=1,j=1}^{n} \quad y_i^T y_j (i,j = 1, \dots, n) \quad (2.16)$$

Untuk mendapatkan jarak tegak lurus yang optimal dengan mempertimbangkan vektor positif, maka hasil perhitungan dari subtitusi nilai x dan nilai y diberi nilai bias =1. Kemudian cari parameter a, dengan terlebih dahulu mencari nilai fungsi setiap tweet, lalu mencari persamaan linear menggunakan persamaan 2.17 dengan memperhatikan i, j =1,…,n berikut.

$$\sum_{i=1,j=1}^{n} a_i T_i^T T_j \quad (2.17)$$

$$\sum_{i=1,j=1}^{n} a_i T_i^T T_j = y_j \quad (2.18)$$

Setelah parameter a didapatkan, kemudian masukkan ke persamaan 2.18 berikut.

$$W = \sum_{i=1}^{n} a_i T_i \quad (2.19)$$

Hasil yang didapatkan menggunakan persamaan, selanjutnya digunakan persamaan 2.19 untuk mendapatkan nilai w dan b yaitu sebagai berikut.

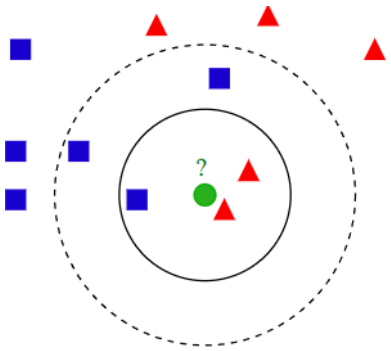
$$y = wx + b \quad (2.20)$$

Sehingga didapatkanlah nilai w dan b atau nilai hyperplane untuk mengklasifikasikan kedua kelas. Pada penelitian ini, kernel yang digunakan adalah kernel liner dalam proses klasifikasi Support Vector Machine.

2.9 K – Nearest Neighbors (K-NN)

Algoritma K - Nearest Neighbor merupakan salah satu teknik yang sangat simpel untuk memecahkan permasalahan klasifikasi. Algoritma ini kerap dipakai untuk klasifikasi teks dan data. K-NN merupakan algoritma yang digunakan buat menerapkan klasifikasi pada sesuatu data objek kedalam kelas yang sudah ditetapkan sebelumnya berdasarkan nilai k-data latih terdekat dengan objek tersebut. Algoritma K - Nearest Neighbor (K-NN) merupakan algoritma supervised learning yang bekerja menggunakan hasil dari model yang baru diklasifikasikan bersumber pada mayoritas dari jenis K-tetangga terdekat.

Tujuan algoritma merupakan untuk mengklasifikasi sebuah model baru berdasarkan atribut serta sampel dari training data. Metode K-NN mempunyai kelebihan yaitu kestabilan dari berapapun variasi nilai-K. hasil yang sudah diperoleh lewat implementasi serta pengujian sistem merupakan jumlah data training, keseimbangan jumlah jenis data training dan nilai K mempengaruhi ketepatan hasil dari analisis sentimen.



Gambar 2.1 Contoh K-NN *Classification*

Mencari jauhnya jarak antar titik pada kelas k akan dihitung memakai jarak Euclidean. Jarak Euclidean merupakan metode pencarian jarak antara dua titik x1 dan x2 yang didefinisikan sebagai berikut :

$$\text{Euclidean} = \sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \qquad (2.21)$$

Keterangan :

Euclidean : jumlah fitur atau dimensi

x : variabel

k : jumlah dimensi variabel

2.10 Confusion Matrix

Confusion Matrix berisi informasi tentang klasifikasi aktual dan yang telah terprediksi yang dilakukan oleh sistem klasifikasi. Umumnya dievaluasi dengan menggunakan data matriks. Metode klasifikasi akan dievaluasi pada bagian akurasi dari hasil klasifikasi. Akurasi sebuah klasifikasi mempengaruhi performa dari suatu klasifikasi. Untuk melakukan analisa dapat digunakan confusion matrix, yaitu sebuah matrik dari prediksi yang akan dibandingkan dengan kelas yang asli dari data inputan[19]. Berikut ini contoh tabel yang menunjukan confusion matrix untuk klasifikasi dua kelas.

Tabel 2.1Confusion Matrix

		Predicted Class	
		Positif	Negatif
Actual Class	Positif	True Positives	False Negatives
	Negatif	False Positives	True Negatives

Keterangan :

True Positives : Merupakan jumlah dokumen dari kelas positif yang benar diklasifikasikan sebagai positif.

False Positives: Merupakan jumlah dokumen dari kelas negatif yang salah diklasifikasikan sebagai positif.

False Negatives : Merupakan jumlah dokumen dari kelas negatif yang benar diklasifikasikan sebagai negatif.

True Negatives: Merupakan jumlah dokumen dari kelas positif yang salah diklasifikasikan sebagai negatif.

Setiap kolom dari tabel confusion matrix merupakan contoh kelas yang telah diprediksi dalam setiap baris mewakili contoh kelas yang sebenarnya. Setelah mendapatkan nilai pada masing-masing kelas, selanjutnya adalah menghitung nilai precision dan akurasinya. Precision adalah ukuran terhadap suatu kelas yang telah diprediksi. Berikut ini adalah persamaannya :

$$Akurasi = \frac{TP-TN}{TP+FP+TN+FN} \quad (2.22)$$

$$Precision = \frac{TP}{TP+FP} \quad (2.23)$$

$$Recall = \frac{TP}{TP+FN} \quad (2.24)$$

dengan TP adalah (True Positives), FP (False Positives), TN (True Negatives), dan FN (False Negatives).

2.11 Penelitian Terkait

Penelitian yang dilakukan oleh Dwi A. P. Rahayu, S. Kuntur, N. Hayatin dengan judul "Sarcasm Detection on Indonesian Twitter Feeds"[6]. Penelitian ini membandingkan dua kombinasi algoritma yaitu pada kombinasi pertama menggunakan Naïve Bayes untuk mengklasifikasi sentimennya lalu dalam mendeteksi sarkasme menggunakan algoritma yang sama yaitu Naïve Bayes, dan pada kombinasi kedua menggunakan Naïve Bayes untuk mengklasifikasi sentimen sedangkan untuk mendeteksi sarkasme menggunakan K - Nearest Neighbors. Pada penelitian ini menggunakan Punctuation, Bag of Words, dan TF-IDF.

Penelitian yang dilakukan oleh E. Indrayubu dengan judul "Komparasi Algoritma Naive Bayes Dan Support Vector Machine Untuk Analisa Sentimen Review Film" [4]. Penelitian ini dalam analisis sentimen pada review film dengan membandingkan algoritma menggunakan Naïve Bayes menghasilkan nilai akurasi yang cukup tinggi 84.50% dengan nilai AUC (Area Under Curve) 0,5000. namun untuk algoritma Support Vector Machine terbukti pada penelitian yang dilakukan menghasilkan nilai akurasi dengan nilai 90.00% dengan nilai AUC sebesar 0.982.

Penelitian yang dilakukan oleh A. Z. Praghakusma dengan judul "Komparasi Fungsi Kernel Metode Support Vector Machine untuk Analisis Sentimen Instagram dan Twitter (Studi Kasus : Komisi Pemberantasan Korupsi)"[20]. Penelitian ini dalam analisis sentimen pada review film dengan membandingkan kernel algoritma SVM menggunakan Linier Kernel, Polynomial Kernel, Sigmoid Kernel menghasilkan nilai akurasi yang berbeda-beda dengan hasil akurasi tertinggi didapat oleh Linier Kernel dengan nilai akurasi 83,06% dan hasil akurasi terkecil 79,83% dengan Sigmoid Kernel.