BAB 2
LANDASAN TEORI

2.1 Piala Dunia U-20

Piala Dunia U-20 merupakan ajang olahraga sepak bola internasional yang dibentuk oleh FIFA *(Federation Internationale de Football Association)* untuk pemain dibawah umur 20 tahun dan diadakan setiap 2 tahun sekali. Piala Dunia U-20 juga dikenal sebagai kejuaraan Dunia FIFA untuk pemain muda. Pengadaan Piala Dunia U-20 pertama kali diadakan pada tahun 1977 dan turnamen ini diadakan di Tunisia [12]. Ketika pertama kali diadakannya kejuaraan ini disebut sebagai Piala Dunia Junior FIFA dan diikuti oleh enam belas tim nasional U-20 dari seluruh dunia. Seiring berjalannya waktu turnamen ini semakin populer dan diikuti lebih banyak negara. Pada tahun 1985, FIFA memutuskan untuk mengubah nama turnamen ini menjadi Kejuaraan Dunia FIFA untuk pemain muda dan menambahkan batasan umur pemain menjadi 20 tahun kebawah. [12].

2.2 Analisa Sentimen

Analisis sentimen merupakan proses memahami, mengekstrak dan mengolah data tekstual untuk mengetahui informasi sentimen yang terkandung dalam suatu kalimat [8]. Analisis sentimen merupakan bagian dari Text mining yang merupakan penelitian komputasi berdasarkan sentimen, emoticon, komentar dan setiap ekspresi yang diungkapkan melalui teks. analisis sentimen dibagi menjadi dua klasifikasi yaitu dokumen klasifikasi pendapat atau fakta dan dokumen klasifikasi ke dalam kelompok negatif, positif dan netral. [13].

2.3 TF-IDF

Metode TF-IDF merupakan metode untuk memberikan nilai bobot pada kata dalam sebuah dokumen atau kalimat. Metode TF-IDF menggabungkan dua konsep dalam perhitungannya, yaitu *term frequency* dan *inversed document frequeny*. *Term Frequency* merupakan frekuensi dari kemunculan sebuah kata didalam suatu dokumen atau opini tertentu sedangkan *inversed document frequency* merupakan frekuensi dari dokumen yang mengandung kata tersebut. Frekuensi dalam kemunculan suatu kata diberikan untuk menunjukan seberapa penting kata

tersebut dalam suatu dokumen. Frekuensi dokumen yang mengandung suatu kata dapat menunjukan seberapa umum kata tersebut. Hal ini menyebabkan hubungan antara sebuah kata dan dokumen akan tinggi apabila frekuensi dari kata tinggi di dalam suatu dokumen atau kalimat dan frekuensi dari keseluruhan dokumen yang mengandung kata tersebut rendah pada kumpulan dokumen [14]. Untuk menghitung TF-IDF dapat menggunakan rumus 2.1, rumus 2.2 dan rumus 2.3.

Rumus perhitungan TF.

$$tf(t,d) = \frac{tf}{max(tf)} \tag{2.1}$$

Rumus perhitungan IDF

$$idf_t = \log(\frac{D}{df_t}) \tag{2.2}$$

Rumus perhitungan TF-IDF

$$W_{t,d} = tf(t,d) \times idf_t \tag{2.3}$$

Keterangan:

tf(t,d) = Frekuensi term (TF).

$max(tf)$ = Total seluruh kata yang ada dalam suatu dokumen.

tf = Jumlah kemunculan term terbanyak. pada dokumen yang sama.

D = Total dari dokumen secara keseluruhan.

$idf(t)$ = bobot kemunculan term t di seluruh dokumen.

$W_{t,d}$ = bobot term dalam suatu dokumen

df_t = jumlah dokumen yang mengandung term t

2.4 *Support Vector Machine*

Support Vector Machine (SVM) merupakan sistem pembelajaran yang menggunakan ruang hipoteses berupa fungsi-fungsi linier dalam sebuah fitur yang memiliki dimensi dan dilatih dengan menggunakan algoritma pembelajaran yang didasarkan pada teori optimasi. *Support Vector Machine* pertama kali dikenalkan oleh Vapnik pada tahun 1992 sebagai rangkaian konsep-konsep unggulan dalam bidang pattern recognition [15].

Metode *Support Vector Machine* dapat digunakan untuk mengklasifikasi data multi kelas. Jika dalam dua dimensi garis pemisah, tiga dimensi berupa

plane, dan dimensi lebih dari tiga maka disebut dengan hyperlane. Dalam beberapa kasus, bentuk dari hyperlane dua feature tidak selalu berbentuk garis pembagi yang lurus. Jika pembagi menggunakan garis lurus maka akan disebut linear, sedangkan jika pembagi tidak berbentuk garis lurus maka akan disebut non-linear seperti polynomial dan RBF [15].

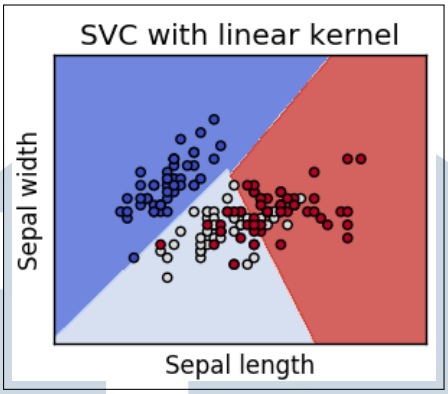
Pada klasifikasi model menggunakan *support vector machine* memiliki beberapa parameter yang berguna untuk meningkatkan kinerja dari pemodelan yaitu *gamma, cost* (C) dan *kernel*. Parameter gamma merupakan parameter yang menentukan seberapa jauh pengaruh dari sample dataset yang dilatih dan Pada parameter gamma nilai rendah berarti jauh sedangkan nilai tinggi berarti dekat, parameter cost (C) merupakan parameter yang digunakan sebagai pengoptimalan metode SVM untuk menghindari kesalahan dalam klasifikasi pada data yang dilatih [16] dan penggunaan kernel Dalam algoritma *support vector machine* berguna untuk mentranformasikan data ke ruang dimensi tinggi [16]. Ada beberapa pilihan fungsi kernel yang dapat digunakan dalam klasifikasi metode *Support Vector Machine* yaitu:

2.4.1 Kernel Linear

Kernel linear merupakan sebuah fungsi sederhana dalam pemodelan SVM. Kernel linear digunakan untuk menganalisis data yang terpisah secara linear [17]. Persamaan fungsi untuk kernel linear dapat dilihat pada persamaan 2.4.

$$K(x,xi) = sum(x * xi) \tag{2.4}$$

Dalam persamaan linear *xi* merupakan data latih dan *x* merupakan data uji *Support Vector Machine*. Dapat dilihat pada Gambar 2.1 garis pemisah kernel linear berbentuk garis lurus.



Gambar 2.1. Kernel Linear

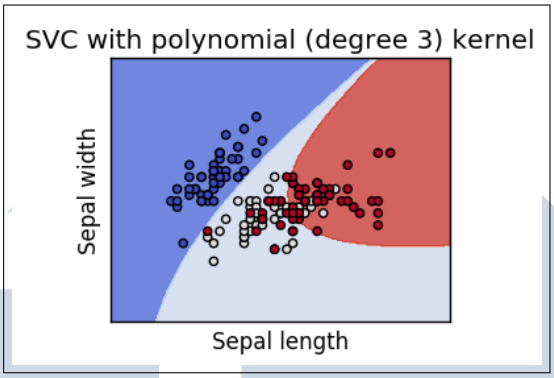
Sumber: [18]

2.4.2 Kernel Polynomial

Kernel polynomial digunakan ketika garis pemisah tidak berbentuk linear. Pada kernel polynomial mewakili kesamaan vektor sample pelatihan dalam ruang dan fitur. Kernel polynomial dapat digunakan untuk memecahkan masalah klasifikasi dataset pelatihan yang sudah dilakukan normalisasi [17]. Persamaan fungsi untuk kernel polynomial dapat dilihat pada persamaan 2.5.

$$K(x,xi) = 1 + sum(x * xi)^d \tag{2.5}$$

Dalam persamaan polynomial *xi* merupakan data latih, *x* merupakan data uji dan derajat (d) merupakan parameter yang berfungsi untuk mencari nilai optimal pada dataset. Semakin besar nilai derajat maka akurasi yang akan dihasilkan kurang stabil. Hal ini terjadi karena semakin tinggi parameternya maka semakin melengkung garis pemisah *hyperlane* yang digunakan. Dapat dilihat dari Gambar 2.2 garis pemisah berbentuk melengkung sesuai dengan derajat yang di input.



Gambar 2.2. Kernel Polynomial

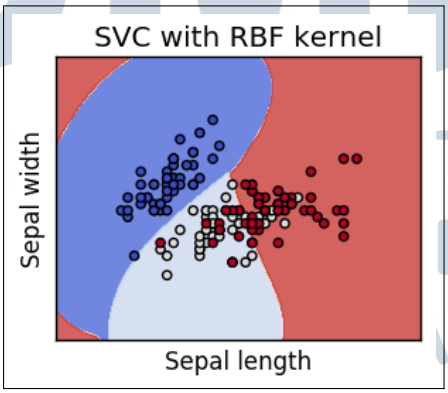
Sumber: [18]

2.4.3 Kernel RBF

Kernel RBF digunakan untuk klasifikasi data yang berbentuk non-linear. Kernel RBF disebut juga dengan kernel Gaussian. Kernel RBF memiliki performa yang baik dengan parameter tertentu, dan juga kernel ini menghasilkan nilai error yang lebih kecil dibanding kernel lain dari data latih [17]. Rumus persamaan kernel RBF dapat dilihat pada persamaan 2.6.

$$K(x,xi) = exp(-gamma * sum((x–xi^2)) \tag{2.6}$$

Dalam persamaan RBF *xi* merupakan titik dari data latih, *x* merupakan data uji dan gamma berfungsi untuk menentukan seberapa jauh pengaruh dari suatu sampel dataset yang dilatih. Dapat dilihat pada Gambar 2.3 pemisah kernel rbf berbentuk non linear.



Gambar 2.3. Kernel RBF

Sumber: [18]

Dalam melakukan perhitungan *Support Vector Machine* metode yang digunakan merupakan *Sequential Minimal Optimization* (SMO) dengan inisialisasi $\alpha = 0$. Berikut adalah langkah langkahn dari metode SMO:

1. Menghitung persamaan dari kernel yang dipilih
2. Menghitung matriks:

$$D_{ij} = Y_i Y_j (x_i x_j) + \lambda^2 \tag{2.7}$$

Keterangan:
D_{ij} = Elemen matriks ke ij.
Y_i = Kelas data ke-i.
Y_j = Kelas data ke-j.
α^2 = Batas teoritis yang diturunkan.

3. Menghitung nilai error

$$E_i = \sum_{j=1}^{n} a_j D_{ij} \tag{2.8}$$

Keterangan:
E_i = Nilai error data ke-i.

4. Menghitung delta a_i

$$\delta a_i = minmax[\gamma(1 - E_i) - a_i], C - a_i \tag{2.9}$$

Keterangan:
δa_i = Delta a ke-i.
γ = Gamma
C = Complexity

5. Menghitung a_i baru

$$a_i baru = a_i + \delta a_i \tag{2.10}$$

Ulangi langkah ke 3 sampai ke 5 hingga kondisi iterasi maksimum tercapai

6. Menghitung nilai $w.x^+$ dan $w.x^-$ untuk mendapatkan nilai bias

$$w.x^+ = a_i Y_i K(w.x^+) w.x^- = a_i Y_i K(w.x^-) b = -1/2(w.x^+ + w.x^-) \tag{2.11}$$

Keterangan:

$w.x^{+}$ = Nilai kernel data x dengan data x positif yang memiliki nilai α tertinggi.

$w.x^{-}$ = Nilai kernel data x dengan data x negatif yang memiliki nilai α tertinggi. b = Nilai bias.

7. Menghitung nilai keputusan

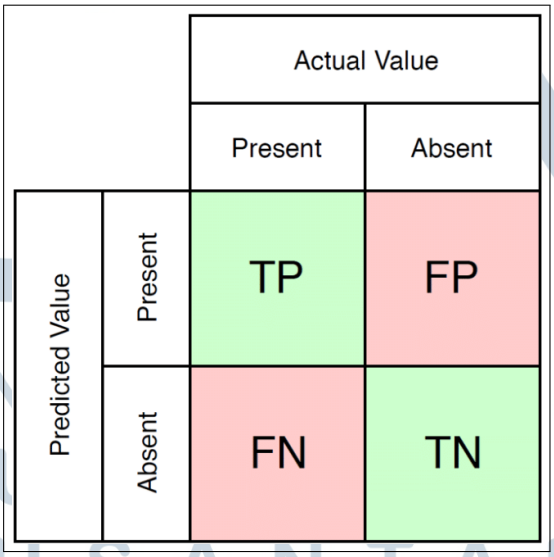
$$f(x) = \sum_{i=1}^{m} sign(a_i y_i K(x, x_i) + b) \tag{2.12}$$

Keterangan:

x = Titik data masukan SVM a_i = nilai bobot setiap titik data $K(x, x_i)$ = fungsi kernel b = nilai bias

2.5 *Confusion Matrix*

Confusion Matrix merupakan pengukuran performa klasifikasi dalam machine learning dapat berupa dua kelas atau lebih. Confusion Matrix memiliki tabel dari empat kombinasi berbeda dari hasil nilai prediksi dan nilai aktual. Contoh dari *Confusion Matrix* dapat dilihat pada Gambar 2.4 [19].



Gambar 2.4. *Confusion Matrix*

Keterangan :

- TP (True Positif): Prediksi yang dibuat positif dan benar.

- TN (True Negatif): Prediksi yang dibuat negatif dan benar
- FP (False Positif): Prediksi yang dibuat positif dan salah
- FN (False Negatif): Prediksi yang dibuat negatif dan salah

Confusion Matrix memiliki beberapa rumus dalam perhitungannya yang digunakan untuk menghitung *accuracy, precision, recall dan F-score*

2.5.1 *Accuracy*

Accuracy dalam Confusion Matrix merupakan perhitungan untuk menggambarkan seberapa akurat model klasifikasi yang telah dibuat dengan benar. Rumus dalam menentukan *accuracy conclusion matrix* dapat dilihat pada rumus 2.13.

$$Accuracy = \frac{TP + TN}{TotalData} \tag{2.13}$$

2.5.2 *Precision*

Precision dalam merupakan rumus untuk menggambarkan akurasi antara data dengan hasil klasifikasi dengan model. Rumus untuk menentukan *precision* dapat dilihat pada rumus 2.14.

$$Precision = \frac{TP}{TP + FP} \tag{2.14}$$

2.5.3 *Recall*

Recall merupakan penggambaran dari keberhasilan model yang diterapkan dalam menemukan kembali sebuah informasi. Pada rumus 2.15 merupakan rumus dari *recall*.

$$Recall = \frac{TP}{TP + FN} \tag{2.15}$$

2.5.4 *F1-score*

F1-score merupakan perbanduingan rata rata dari hasil *precision* dan *recal* yang telah dibobotkan. Untuk rumus *f1-socer* dapat dilihat pada rumus 2.16.

$$F1-Score = 2 \times \frac{precision \times recall}{precision + recall} \tag{2.16}$$