BAB III
LANDASAN TEORI

3.1. Facebook

Facebook adalah sebuah layanan jejaring sosial berkantor pusat di Menlo Park, California, Amerika Serikat, yang diluncurkan pada bulan Februari 2004. Riset yag dilakukan *We Are Social*, hingga April 2017, Facebook memiliki hampir dua milyar pengguna aktif dan lebih dari separuhnya menggunakan telepon genggam (Kemp, 2017).

Facebook didirikan oleh Mark Zuckerberg bersama teman sekamarnya dan sesama mahasiswa Universitas Harvard, Eduardo Saverin, Andrew McCollum, Dustin Moskovitz, dan Chris Hughes. Keanggotaan situs web ini awalnya terbatas untuk mahasiswa Harvard saja, kemudian diperluas ke perguruan lain di Boston, Ivy League, dan Universitas Stanford. Situs ini secara perlahan membuka diri kepada mahasiswa di universitas lain sebelum dibuka untuk siswa sekolah menengah atas, dan akhirnya untuk setiap orang yang berusia minimal 13 tahun (Carlson, 2010).

Pengguna facebook dapat membuat profil dilengkapi foto, daftar ketertarikan pribadi, informasi kontak, dan informasi pribadi lain. Pengguna dapat berkomunikasi dengan teman dan pengguna lain melalui pesan pribadi atau umum dan fitur obrolan. Mereka juga dapat membuat dan bergabung dengan grup ketertarikan dan "halaman kesukaan", beberapa di antaranya diurus oleh banyak organisasi dengan maksud beriklan (*Facebook*, 2008).

3.1.1 Facebook API Graf

Facebook API adalah platform untuk membangun aplikasi yang tersedia bagi anggota jejaring sosial *Facebook*. API memungkinkan aplikasi untuk menggunakan koneksi sosial dan informasi profil untuk membuat aplikasi lebih melibatkan, dan mempublikasikan aktivitas ke *feed* berita dan halaman profil Facebook, sesuai dengan pengaturan privasi pengguna individual. Dengan API, pengguna dapat menambahkan konteks sosial ke

aplikasinya dengan memanfaatkan data profil, teman, halaman, grup, foto, dan acara. API menggunakan protokol dan tanggapan yang tenang dalam format JSON (Berlind, 2009).

API Graf merupakan cara utama untuk mendapatkan data dari, dan memasukkan data ke, platform Facebook. API berbasis HTTP dengan level rendah yang dapat digunakan untuk mengkueri data secara terprogram, mengirimkan cerita baru, mengelola iklan, mengunggah foto, dan melakukan berbagai jenis tugas lain yang mungkin diterapkan aplikasi.

3.2. *Unstructured Data*

Data diklasifikasikan dalam 2 jenis, yaitu terstruktur dan tidak terstruktur. Data terstruktur direpresentasikan dalam skema yang jelas sehingga mudah untuk dianalisis maupun diintegrasikan dengan data terstruktur lainnya. Sedangkan data tidak terstruktur direpresentasikan dalam berbagai bentuk sehingga sangat sulit untuk dianalisa maupun diintegrasikan dengan sumber data lain. Data terstruktur dan tidak terstruktur apabila diintegrasikan dalam menganalisis suatu permasalahan akan memberikan pemahaman dan solusi yang lebih lengkap dan tepat sasaran. Namun, bukanlah hal yang mudah untuk mengintegrasikan kedua jenis data tersebut. Dibutuhkan teknik maupun arsitektur yang tepat untuk mengatasi permasalahan tersebut (Afifanto, 2015).

Data tidak terstruktur adalah informasi yang sulit dihitung dan diatur dalam *spreadsheet* serta tidak mudah dianalisis, namun berpotensi menghasilkan hasil prediksi yang kuat jika digunakan bersamaan melalui analisis teks. Contoh data tidak terstruktur meliputi cerita atau esai, jawaban tertulis atas pertanyaan tanggapan bebas, atau ulasan karyawan perusahaan di media sosial seperti: *facebook, twitter, instagram*. Hal-hal yang mendorong pertambahan jumlah data tidak terstruktur diantaranya *mobile sensors*, social media, *video surveillance, video rendering, smart grids, geophysical exploration, medical imaging* (Wiryanto, 2017).

3.3. *Web Crawling*

Web crawling adalah proses yang digunakan untuk mengekstrak data dari situs web dengan mengakses *World Wide Web* secara langsung menggunakan *HTML* melalui *web browser*. Halaman web dibangun dengan menggunakan bahasa *markup* berbasis teks (*HTML*), dan sering mengandung sejumlah data besar yang berguna dalam bentuk teks. Perusahaan seperti Facebook menyediakan alat untuk melakukan *crawling* yaitu API atau alat untuk mengekstrak data dari situs web (Boeing, 2016).

3.4. Shopee



Gambar 3.1 *Logo Shopee (Shopee, 2015)*

Shopee merupakan platform *marketplace online* yang menjembatani penjual dan pembeli untuk mempermudah transaksi jual beli *online* melalui perangkat ponsel mereka. Shopee menawarkan berbagai macam produk-produk mulai dari produk *fashion* sampai dengan produk untuk kebutuhan sehari-hari. Sasaran pengguna Shopee adalah kalangan muda yang saat ini terbiasa melakukan kegiatan dengan bantuan gawai termasuk kegiatan berbelanja. Untuk itu Shopee hadir dalam bentuk aplikasi *mobile* guna untuk menunjang kegiatan berbelanja yang mudah dan cepat. Shopee hadir dalam bentuk aplikasi *mobile* untuk memudahkan penggunanya dalam melakukan kegiatan belanja *online* tanpa harus membuka website melalui perangkat komputer.

Shopee sendiri merupakan perpanjangan tangan dari Garena untuk merambah ke segmen *e-commerce*. Garena adalah penyedia platform internet konsumen yang berbasis di Asia yang didirikan di Singapura pada tahun 2009. Shopee tidak hanya hadi di pasar Indonesia saja, tetapi telah hadir di pasar

Malaysia, Thailand, Singapura, Filipina, Taiwan, dan juga Vietnam. Merujuk pada halaman *Facebook* mereka, Shopee masuk ke pasar Indonesia pada bulan Mei 2015 dan mulai beroperasi pada akhir Juni 2015 (Priambada, 2015).

Keputusan membuka cabang di Indonesia sendiri tak terlepas dari potensi yang dipunyai negara ini. CEO Shopee Chris Feng mengatakan Indonesia memiliki jumlah penduduk yang besar dengan angka lebih dari 250 juta. Selain itu, penetrasi internet di Tanah Air makin tinggi dengan lebih dari 100 juta pengguna internet. Meski dibuka di tujuh negara sekaligus, Shopee diberikan sentuhan lokal, termasuk di Indonesia. Menurut Chris, setiap negara memiliki karakteristik sendiri. Untuk itu, ia lebih banyak merekrut tenaga profesional lokal agar mengerti cita rasa penduduk setempat. Shopee memfokuskan sendiri sebagai *marketplace* berbasis *mobile*. Kata Chris, platform *mobile* dirasa lebih mudah dan banyak digunakan (Rahman, 2015).

Beberapa keunggulan Shopee menurut Chris Feng, CEO Shopee dalam acara peluncuran Shopee di Jakarta adalah:

1. Menjual barang dengan cukup cepat, dapat dilakukan hanya dalam waktu 30 detik.
2. Memiliki tampilan yang sederhana dan dapat dengan mudah digunakan walaupun oleh pengguna baru.
3. Menawarkan fitur *chatting* dengan penjual sehingga dengan mudah bisa langsung bertransaksi atau tawar menawar. Fitur ini sangat memudahkan pengguna aplikasi Shopee mengingat di aplikasi atau situs *e-commerce* lainnya pembeli harus menyimpan nomor telepon terlebih dahulu untuk berhubungan langsung. Fitur *chatting* pada aplikasi Shopee juga berbeda karena selain bisa mengirim pesan tetapi juga bisa mengirim tautan seperti foto.
4. Memiliki fitur tawar yang memungkinkan pembeli untuk menawar harga yang sudah ditetapkan oleh penjual.
5. Memiliki fitur berbagi yang lengkap sehingga dengan mudah menyebarkan info ke berbagai media *social* atau *messenger*, seperti Facebook, Instagram, Twitter, Line, Pinterest, Whatsapp, Buzz, hingga Beetalk.

6. Shopee mengintegrasikan fitur media sosial mencakup fungsi hashtag, yang memungkinkan pengguna mencari barang atau produk yang sedang populer atau untuk mengikuti tren produk terbaru dengan mudah. (Reza, 2015)

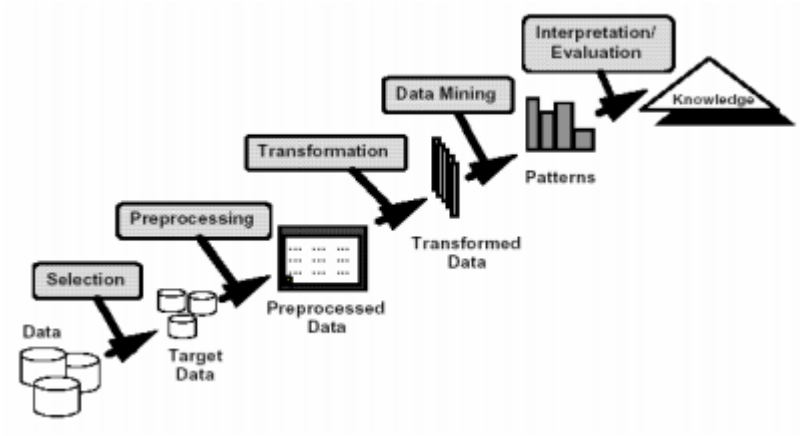
3.5. Analisis Deskriptif

Analisis deskriptif adalah analisis yang digunakan untuk meringkas dan menyajikan data dalam bentuk tabel, grafik atau ringkasan numerik data. Hasan (2001) menjelaskan bahwa statistik deskriptif adalah bagian dari statistika yang mempelajari cara pengumpulan data dan penyajian data sehingga mudah dipahami. Statistika deskriptif hanya berhubungan dengan hal menguraikan atau memberikan keterangan-keterangan mengenai suatu data atau keadaan. Dengan kata lain statistika deskriptif berfungsi menerangkan keadaan, gejala, atau persoalan.

Subagyo (2003) menyatakan yang dimaksud sebagai statistika deskriptif adalah bagian statistika mengenai pengumpulan data, penyajian, penentuan nilai-nilai statistika, pembuatan diagram atau gambar mengenai data yang disajikan dalam bentuk yang lebih mudah dipahami atau dibaca.

3.6. *Data Mining*

Data mining adalah kegiatan menemukan pola yang menarik dari data dalam jumlah besar, data dapat disimpan dalam *database*, data *warehouse*, atau penyimpanan informasi lainnya. *Data mining* berkaitan dengan bidang ilmu – ilmu lain, seperti *database system*, *data warehousing*, statistik, *machine learning*, *information retrieval*, komputasi tingkat tinggi dan data visualisasi. Selain itu, *data mining* didukung oleh ilmu lain seperti *neural network*, pengenalan pola, *spatial data analysis, image database, signal processing. Data mining* disebut juga *knowledge discovery (mining) in databases (KDD), knowledge extraction, data/pattern analysis, data archeology, data dredging, information harvesting, business intelligence,* dan sebagainya. (Han, 2006).



Gambar 3.2 *Tahapan dalam proses KDD (Berry dan Linoff, 1999)*

Dalam aplikasinya, *data mining* sebenarnya merupakan salah satu bagian proses *Knowledge Discovery in Database (KDD)* yang bertugas untuk mengekstrak pola atau model dari data dengan menggunakan suatu algoritma yang spesifik. Adapun proses KDD (Berry & Linoff, 1999) sebagai berikut:

1. *Data selection*: pemilihan data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai.
2. *Preprocessing*: sebelum proses data mining dapat dilaksanakan, perlu dilakukan proses *cleaning* dengan tujuan untuk membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak (tipografi). Juga dilakukan proses enrichment, yaitu proses "memperkaya" data yang sudah ada dengan data atau informasi lain yang relevan dan diperlukan untuk KDD, seperti data atau informasi eksternal.
3. *Transformation*: yaitu proses *coding* pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses *data mining*. Proses *coding* dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam *database*.
4. *Data mining*: proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu.
5. *Interpretation / Evaluation*: pola informasi yang dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak

yang berkepentingan. Tahap ini merupakan bagian dari proses KDD yang disebut dengan *interpretation*. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesa yang ada sebelumnya atau tidak.

Dalam proses KDD yang sesungguhnya, dapat saja terjadi iterasi atau pengulangan pada tahap tahap tertentu. Pada setiap tahap dalam proses KDD, seorang analis dapat saja kembali ke tahap sebelumnya. Sebagai contoh, pada saat *coding* atau *data mining*, analis menyadari proses *cleaning* belum dilakukan dengan sempurna, atau mungkin saja analis menemukan data atau informasi baru untuk "memperkaya" data yang sudah ada.

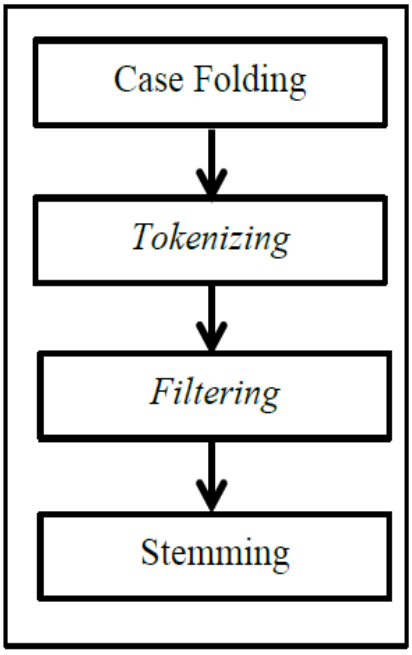
3.7. *Machine Learning*

Machine learning adalah cabang dari kecerdasan buatan, merupakan disiplin ilmu yang mencakup perancangan dan pengembangan algoritma yang memungkinkan komputer untuk mengembangkan perilaku yang didasarkan kepada data empiris, seperti dari sensor data pada basis data. Sistem pembelajaran dapat memanfaatkan contoh (data) untuk menangkap ciri yang diperlukan dari probabilitas yang mendasarinya (yang tidak diketahui). Data dapat dilihat sebagai contoh yang menggambarkan hubungan antara variabel yang diamati. Karena itu *machine learning* harus merampatkan (generalisasi) perilaku dari contoh yang ada untuk menghasilkan keluaran yang berguna dalam kasus-kasus baru (Wikipedia, 2013).

3.8. *Text Mining*

Text mining adalah proses ekstraksi pola (informasi dan pengetahuan yang berguna) dari sejumlah besar sumber data tak terstruktur. *Text mining* memiliki tujuan dan menggunakan proses yang sama dengan *data mining*, namun memiliki masukan yang berbeda. Masukan untuk *text mining* adalah adalah data yang tidak (atau kurang) terstruktur, seperti dokumen Word, PDF, kutipan teks, dan lain-lain, sedangkan masukan untuk *data mining* adalah data yang terstruktur (Feldman, 2007).

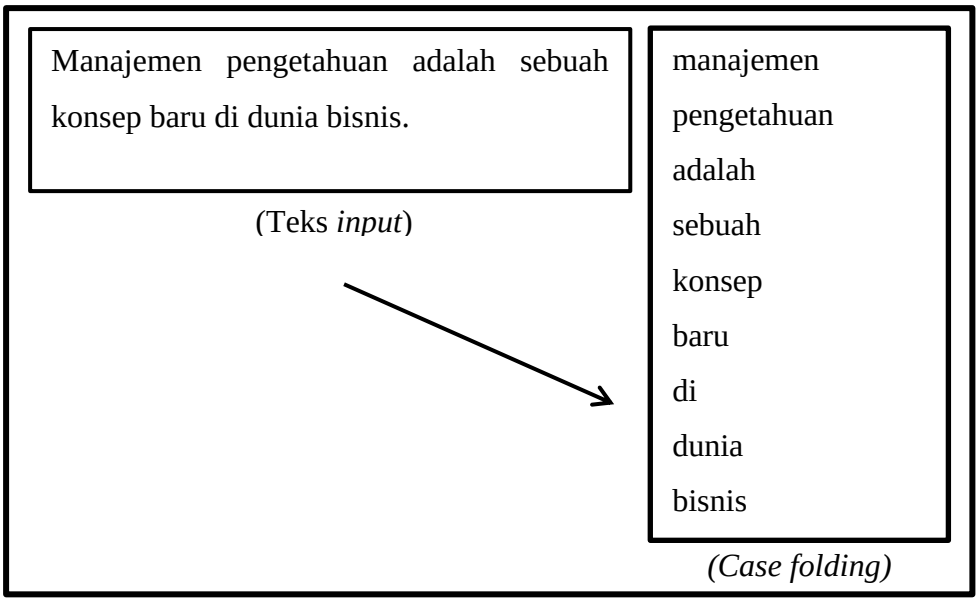
Teks yang dilakukan proses *text mining*, pada umumnya memiliki beberapa karakteristik diantaranya adalah memiliki dimensi yang tinggi, terdapat *noise* pada data, dan terdapat struktur teks yang tidak baik. Cara yang digunakan dalam mempelajari struktur data teks adalah dengan terlebih dahulu menentukan fitur-fitur yang mewakili setiap kata untuk setiap fitur yang ada pada dokumen. Sebelum menentukan fitur-fitur yang mewakili, diperlukan tahap *pre-processing* yang dilakukan secara umum dalam *text mining* pada dokumen, yaitu *case folding*, *tokenizing*, *filtering*, dan *stemming* (Mooney, 2006), seperti terlihat pada gambar 3.3 berikut:



Gambar 3.3 *Proses text mining (Mooney, 2006)*

3.8.1. *Case Folding*

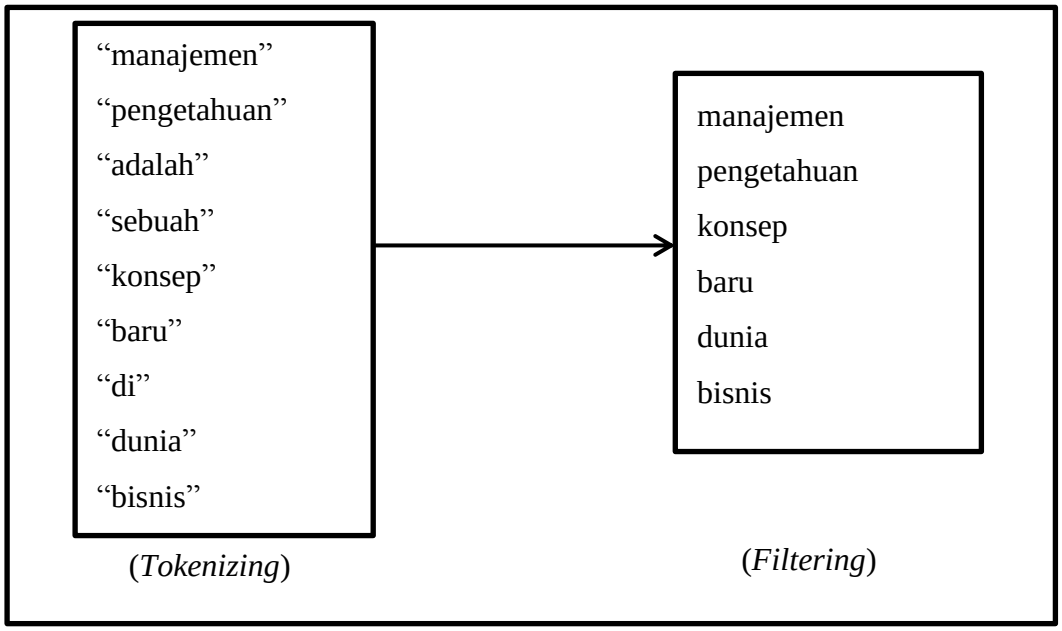
Case folding adalah mengubah semua huruf kapital dalam dokumen menjadi huruf kecil, hanya huruf ‘a’ sampai dengan ‘z’ yang diterima. Karakter selain huruf dihilangkan dan dianggap sebagai delimeter. Proses *case folding* seperti pada gambar 3.4 berikut:



Gambar 3.4 *Proses case folding (Ulina, 2014)*

3.8.2. *Tokenizing* dan *Filtering*

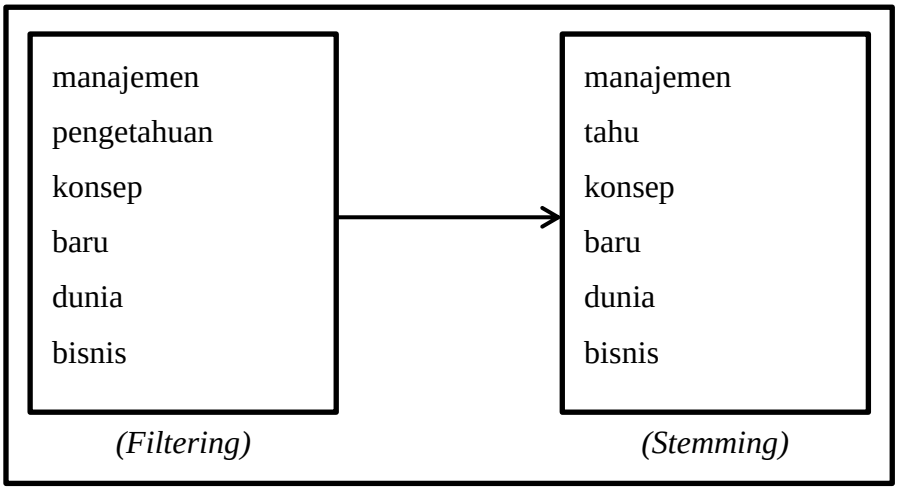
Tokenizing adalah proses pemotongan *string input* di dalam teks berdasarkan kata-kata yang menyusunnya. Hasil dari *tokenizing* adalah seluruh kata-kata yang telah dipisahkan dalam suatu kalimat. Sedangkan *filtering* adalah proses pengambilan kata-kata yang penting atau yang dibutuhkan untuk dianalisis. Proses *tokenizing* dan *filtering* seperti pada gambar 3.5 berikut:



Gambar 3.5 *Proses tokenizing dan filtering (Ulina, 2014)*

3.8.3. *Stemming*

Tahap *stemming* adalah tahap mencari kata dasar dari setiap kata yang mengandung imbuhan (deduktif dan induktif). Pada tahap ini berbagai bentukan kata yang mengandung imbuhan akan memiliki representasi yang sama dengan kata yang tidak mengandung imbuhan. Proses *stemming* pada teks berbahasa Indonesia dijelaskan pada gambar 3.6 berikut:



Gambar 3.6 *Proses stemming (Ulina, 2014)*

3.9. Analisis Sentimen

Analisis sentimen adalah bidang studi yang menganalisis pendapat, sentimen, evaluasi, sikap, dan emosi seseorang dari bahasa tulisan. Analisis sentimen semakin penting bertepatan dengan pertumbuhan media sosial seperti ulasan, diskusi forum, blog, blog mikro, Twitter, dan jejaring sosial (Liu, 2012). Tugas dasar dalam analisis sentimen adalah mengelompokkan polaritas dari teks yang ada dalam dokumen, kalimat, atau pendapat. Polaritas mempunyai arti apakah teks yang ada dalam dokumen, kalimat, atau pendapat memiliki aspek positif atau negatif (Ling, et al., 2014).

Klasifikasi sentimen biasanya digunakan untuk menyelesaikan masalah klasifikasi dua kelas, positif dan negatif. Data pengujian yang digunakan biasanya adalah ulasan produk secara *online*. Karena ulasan *online* memiliki nilai penilaian yang ditetapkan oleh para penelaah, misalnya kelas positif dan negatif ditentukan menggunakan peringkat bintang 1 sampai 5, ulasan dengan bintang 4 atau 5

dianggap sebagai tinjauan positif, dan ulasan dengan bintang 1 sampai 2 dianggap sebagai ulasan negatif. Sebagian besar makalah penelitian tidak menggunakan kelas netral, dengan tujuan agar memudahkan masalah klasifikasi, namun juga tidak jarang peneliti yang menggunakan kelas netral, misalnya penggunaan peringkat bintang 3 sebagai kelas netral (Liu, 2012).

3.10. Pembobotan Kata

Hal yang perlu diperhatikan dalam pencarian informasi dari koleksi dokumen yang heterogen adalah pembobotan *term*. *Term* dapat berupa kata, frase atau unit hasil *indexing* lainnya dalam suatu dokumen yang dapat digunakan untuk mengetahui konteks dari dokumen tersebut. Karena setiap kata memiliki tingkat kepentingan yang berbeda dalam dokumen, maka untuk setiap kata tersebut diberikan sebuah indikator, yaitu *term weight* (Zafikri, 2008).

Menurut Mandala (dalam Zafikri, 2008) *term weighting* atau pembobotan *term* sangat dipengaruhi oleh hal-hal berikut ini :

1. *Term Frequency* (TF) *factor*, yaitu faktor yang menentukan bobot *term* pada suatu dokumen berdasarkan jumlah kemunculannya dalam dokumen tersebut. Nilai jumlah kemunculan suatu kata (*term frequency*) diperhitungkan dalam pemberian bobot terhadap suatu kata. Semakin besar jumlah kemunculan suatu *term* (tf tinggi) dalam dokumen, semakin besar pula bobotnya dalam dokumen atau akan memberikan nilai kesesuian yang semakin besar.
2. *Inverse Document Frequency* (IDF) *factor*, yaitu pengurangan dominansi *term* yang sering muncul di berbagai dokumen. Hal ini diperlukan karena *term* yang banyak muncul di berbagai dokumen, dapat dianggap sebagai *term* umum (*common term*) sehingga tidak penting nilainya. Sebaliknya faktor kejarangmunculan kata (*term scarcity*) dalam koleksi dokumen harus diperhatikan dalam pemberian bobot. Menurut Mandala, kata yang muncul pada sedikit dokumen harus dipandang sebagai kata yang lebih penting (*uncommon terms*) daripada kata yang muncul pada banyak dokumen. Pembobotan akan memperhitungkan faktor kebalikan frekuensi dokumen yang mengandung suatu kata (*Inverse Document Frequency).*

3.10.1 Pembobotan *Term Frequency-Inverse Document Frequency*

Adapun metode pembobotan yang paling banyak digunakan adalah metode TF-IDF (*Term Frequency* dengan *Inverse Document Frequency*). Perhitungan bobot pada metode ini dilakukan dengan mengalikan nilai yang dihasilkan oleh TF dan IDF). Menurut Defeng (dalam Rijaalul Fattah, 2016), jenis formula yang akan digunakan untuk perhitungan *term frequency* (TF) yaitu TF murni (*raw* TF). TF menunjukkan seberapa sering sebuah kata muncul dalam suatu dokumen tertentu. Sedangkan IDF menjelaskan tentang seberapa penting arti sebuah kata dalam sebuah dokumen yang dirumuskan sebagai berikut:

$$IDF(A) = \log \frac{D}{df_A} \tag{3.1}$$

Adapun nilai D sama dengan df_j maka akan didapatkan hasil nol untuk perhitungan IDF sehingga perlu ditambahkan nilai 1(+1) untuk rumus IDF menjadi:

$$IDF(A) = \log \frac{D}{df_A} + 1 \tag{3.2}$$

$$W(A) = TF(A) + IDF\ (A) \tag{3.3}$$

dengan

$W(A)$ = bobot *term* dari kata A

$TF(A)$ = jumlah kata A dalam dokumen (term frequency)

D = jumlah seluruh dokumen

df_A = jumlah dokumen yang memuat kata A di dalamnya

3.11. *Support Vector Machine*

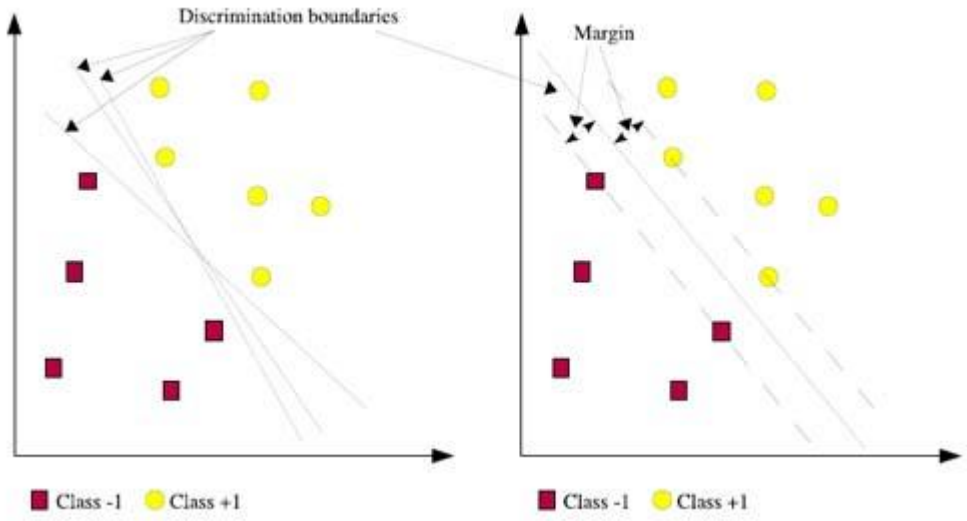
Support Vector Machine (*SVM*) merupakan metode yang dikembangkan oleh Boser, Guyon, Vapnik, dan dipresentasikan pertama kali pada tahun 1992 pada acara *Annual Workshop on Computational Learning Theory*. Konsep dasar *SVM* sebenarnya merupakan kombinasi harmonis dari teori-teori komputasi yang telah ada puluhan tahun sebelumnya, kernel diperkenalkan oleh Aronszajn tahun 1950, dan demikian juga dengan konsep-konsep pendukung yang lain. Akan tetapi

hingga tahun 1992, belum pernah ada upaya merangkaikan komponen-komponen tersebut.

Support Vector Machine (SVM) adalah sistem pembelajaran yang menggunakan ruang hipotesis berupa fungsi-fungsi linier dalam sebuah ruang fitur (*feature space*) berdimensi tinggi, dilatih dengan algoritma pembelajaran yang didasarkan pada teori optimasi dengan mengimplementasikan learning bias yang berasal dari teori pembelajaran statistik (Santoso, 2007).

Konsep klasifikasi dengan *Support Vector Machine* (*SVM*) adalah dengan cara mencari *hyperplane* terbaik untuk memisahkan dua kelas data. *SVM* bekerja dengan memaksimalkan margin yang merupakan jarak pemisah antara kedua kelas data tersebut.

3.11.1. *Support Vector Machine* pada Data Terpisah Secara Linier



Gambar 3.7 *Ilustrasi SVM menemukan hyperplane terbaik yang memisahkan dua kelas -1 dan +1*

Pada gambar 3.7 memperlihatkan beberapa *pattern* yang berasal dari kelas +1 (positif) dan -1 (negatif). *Pattern* pada kelas +1 diberi simbol bentuk lingkaran berwarna kuning sedangkan *pattern* pada kelas -1 disimbolkan dengan kotak berwarna merah. Pengklasifikasian menggunakan *SVM* dilakukan dengan usaha menemukan *hyperplane* terbaik yang bisa

memisahkan kedua kelas tersebut. *Margin hyperplane* dengan *pattern* terdekat dari masing-masing kelas diukur untuk dapat menemukan *hyperplane* pemisah terbaik untuk kedua kelas tersebut. *Pattern* terdekat ini dinamakan *support vector*. Garis sebelah kanan menunjukan *hyperplane* terbaik yang berada di tengah antara kedua kelas. Sedangkan pada jalur garis putus-putus terdapat simbol lingkaran warna kuning dan simbol kotak warna merah yang berperan sebagai *support vector*.

Data yang tersedia dinotasikan sebagai $x \in R^d$ sedangkan label masing-masing kelas dinotasikan $y_i \ni \{-1, +1\}$ untuk $i = 1,2,3, \ldots n$.

Diasumsikan kedua kelas dapat terpisah secara sempurna oleh *hyperplane* berdimensi *d,* yang didefinisikan:

$$\vec{w}.\vec{x} + b = 0 \quad (3.4)$$

Pattern $\vec{x}_i$ yang *term*asuk kelas -1 (sampel negatif) dapat dirumuskan sebagai *pattern* yang memenuhi pertidaksamaan:

$$\vec{w}.\vec{x} + b \leq -1 \quad (3.5)$$

Sedangkan pattern $\vec{x}_i$ yang *term*asuk kelas +1 (sampel positif)

$$\vec{w}.\vec{x} + b \geq +1 \quad (3.6)$$

dengan

R^d = ruang vektor ,d > 1

n = banyak data

$\vec{w}$ = vektor bobot

$\vec{x}$ = vektor data (*input*)

b = bias

Margin terbesar dapat ditemukan dengan memaksimalkan nilai jarak antara jarak dan titik terdekatnya, yaitu $\frac{1}{\|w\|}$. Hal ini dapat dirumuskan sebagai *Quadratic Programming* (QP) problem, yaitu mencari titik minimal Persamaan 3.7, dengan memperlihatkan *constraint* Persamaan 3.8

$$min_w = \tau(w) = \frac{1}{2}\|w\|^2 \quad (3.7)$$

$$y_i(x_i.w + b) - 1 \geq 0 \quad (3.8)$$

dengan ||w|| adalah vektor normal.

Masalah ini dapat dipecahkan dengan berbagai teknik komputasi, diantaranya *lagrange multiplier* yang dinyatakan pada Persamaan 3.9.

$$L(w, b, \alpha) = \frac{1}{2\|w\|^2} 2 - \sum \alpha_i (y_i((\vec{x}.\vec{w} + b) - 1)) \quad (3.9)$$

dengan i = 1,2, … , n.

Di mana α adalah *lagrange multiplier*, yang bernilai 0 atau positif $a_i \geq 0$. Nilai optimal dari Persamaan 3.9 dapat dihitung dengan meminimalkan L terhadap w dan b, dan memaksimalkan L terhadap a_i, dengan memperhatikan sifat bahwa pada titik optimal gradien L = 0. Persamaan 3.9 dapat dimodifikasi sebagai maksimalisasi *problem* yang hanya mengandung a_i, sebagaimana terlihat pada Persamaan 3.10 dan Persamaan 3.11 dibawah ini.

$$\sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} a_i \, a_j y_i y_j \vec{x}_i \vec{x}_j \quad (3.10)$$

di mana $\alpha_i \geq 0 (i = 1,2, \ldots . l) \sum_{i=1}^{l} \alpha_i \, y_i = 0$ (3.11)

Dengan demikian, maka akan diperoleh a_i yang kebanyakan bernilai positif yang disebut sebagai *support vector* dan juga memperoleh Persamaan 3.12 dan Persamaan 3.13 sebagai solusi bidang pemisah.

$$w = \sum \alpha_i \, y_i x_i \quad (3.12)$$

$$b = y_k - w^T x_k \quad (3.13)$$

3.12. *Wordcloud*

Word cloud adalah salah satu hasil dari metode *text mining* yang menampilkan kata-kata populer terkait dengan kata kunci internet dan data teks. Menurut PBC (dalam Arkhamsiagustinah, 2015) *word cloud* sering digunakan untuk menyoroti istilah populer atau *trend* berdasarkan frekuensi pengguna. Menurut Shawn Graham, Ian Milligan, dan Scott Weingart (dalam Arkhamsiagustinah, 2015) *word cloud* merupakan pendekatan yang dapat menjelaskan pertanyaan penelitian dengan sangat cepat dan mudah, dapat menjelajahi *word cloud* secara singkat dan dapat melakukan analisis yang komprehensif. Kata yang paling sering muncul di dalam data teks akan memiliki bentuk yang paling besar, begitu pula sebaliknya.

3.13. Asosiasi Kata

Istilah korelasi sering digunakan untuk menyatakan hubungan dua atau lebih variabel yang sifatnya kuantitatif, sedangkan istilah asosiasi sering dimaknai keeratan hubungan antara dua atau lebih variabel yang sifatnya kualitatif (Ulwan, 2016). Asosiasi merupakan proses pencarian hubungan antar elemen data. Dalam dunia industri retail, analisis asosiasi biasanya disebut *Market Basket Analysis* (Miner et al, 2012). Nilai asosiasi didapatkan dengan cara menghitung nilai korelasi menggunakan rumus berikut:

$$r_{xy} = \frac{n\sum x_i y_i - (\sum x_i)(\sum y_i)}{\sqrt{\{n\sum x_i^2 - (\sum x_i)^2\}\{n\sum y_i^2 - (\sum y_i)^2\}}} \qquad (3.14)$$

dengan

r_{xy} = nilai korelasi antara variabel x dan variabel y

n = banyaknya pasangan data x dan y

$\sum x_i$ = jumlah nilai pada variabel x $\qquad i = 1,2,3, \dots \dots, n$

$\sum y_i$ = jumlah nilai pada variabel y

$\sum x_i^2$ = kuadrat dari total nilai variabel x

$\sum y_i^2$ = jumlah nilai pada variabel y

$\sum x_i y_i$ = jumlah dari hasil dari perkalian antara variabel x dan variabel y