# BAB 2
# TINJAUAN PUSTAKA

## 2.1 *Data Mining*

Pengertian *data mining* ialah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam *database* [10]. *Data mining* adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai *database* besar [11]. Menurut Gartner Group, *data mining* adalah proses menemukan hubungan baru yang mempunyai arti, pola dan kebiasaan dengan memilah-milah sebagian besar data yang disimpan dalam media penyimpanan dengan menggunakan teknologi pengenalan pola seperti teknik statistik dan matematika. *Data mining* merupakan gabungan dari beberapa disiplin ilmu yang menyatukan teknik dari pembelajaran mesin, pengenalan pola, statistik, *database*, dan visualisasi untuk penanganan permasalahan pengambilan informasi dari *database* yang besar.

*Data mining* secara umum dibagi menjadi beberapa metode, berikut ini adalah metode-metodenya:

### 2.1.1 *Association*

*Association* juga disebut sebagai *Market Basket Analysis* adalah metode berbasis aturan yang digunakan untuk menemukan asosiasi dan hubungan variabel dalam satu set data. Biasanya analisis ini terdiri dari pernyataan "*if* atau *then*" sederhana. *Association* banyak digunakan dalam mengidentifikasi korelasi produk dalam keranjang belanja untuk memahami kebiasaan konsumsi pelanggan. Sehingga, perusahaan dapat mengembangkan strategi penjualan dan membuat sistem rekomendasi yang lebih baik

### 2.1.2 *Classification*

*Classification* adalah metode yang paling umum pada *Data mining*. Persoalan bisnis seperti Churn Analysis, dan *Risk Management* biasanya melibatkan metode *Classification*. *Classification* adalah tindakan untuk memberikan kelompok pada setiap keadaan. Setiap keadaan berisi sekelompok atribut, salah satunya adalah *class attribute*. Metode ini butuh untuk menemukan sebuah model yang dapat menjelaskan *class attribute* itu sebagai fungsi dari *input attribute*.

### 2.1.3 *Regression*

*Regression* adalah teknik yang menjelaskan variabel dependen melalui proses analisis variabel independen. Sebagai contoh, prediksi penjualan suatu produk berdasarkan korelasi antara harga produk dengan tingkat pendapatan rata-rata pelanggan. Teknik paling populer yang digunakan untuk *regression* adalah *linear regression* dan *logistic regression*. Teknik lain yang didukung oleh SQL Server *Data mining* adalah *Regression Trees* (bagian dari algoritma Microsoft Decission Trees) dan Neural Network.

### 2.1.4 *Clustering*

*Clustering* juga disebut sebagai *segmentation*. Metode ini digunakan untuk mengidentifikasi kelompok alami dari sebuah kasus yang didasarkan pada sebuah kelompok atribut, mengelompokkan data yang memiliki kemiripan atribut. *Clustering* adalah metode *Data mining* yang *unsupervised*, karena tidak ada satu atribut pun yang digunakan untuk memandu proses pembelajaran, jadi seluruh atribut *input* diperlakukan sama.

Kebanyakan Algoritma *Clustering* membangun sebuah model melalui serangkaian pengulangan dan berhenti ketika model tersebut telah memusat atau berkumpul (batasan dari segmentasi ini telah stabil).

## 2.2 *Text Mining*

*Text Mining* adalah proses ekstraksi pengetahuan yang digunakan pengguna berinteraksi dan bekerja dengan sekumpulan dokumen yang berisi teks menggunakan beberapa menggunakan beberapa alat analisis [8]. Tujuan dari *Text*

*Mining* adalah untuk menentukan pola dan menemukan informasi baru informasi sehingga dapat digunakan untuk memproses, mengatur, dan menganalisis teks yang tidak terstruktur dalam jumlah besar [8]. Dalam prosesnya *Text Mining* menggabungkan teknik *Data mining, machine learning, natural language processing, information retrival* dan *knowledge management.*

## 2.3 *Preprocessing*

*Preprocessing* merupakan tahapan pra-pemrosesan untuk meningkatkan kualitas analisis terhadap tweet yang menjadi fokus penelitian. Tahapan *Preprocessing* penting dilakukan dalam menyusun teks yang tidak terstruktur serta menjaga *keyword* yang dapat berguna untuk mewakili topik [12]. Dalam tahap *Preprocessing*, terdapat beberapa metode yang dipakai yaitu :

### 2.3.1 *Case folding*

*Case folding* merupakan proses mengubah seluruh karakter pada tweet menjadi huruf kecil atau *lowercase* [13]. Dengan mengubah seluruh karakter menjadi huruf kecil, kita memudahkan analisis teks dengan mengabaikan perbedaan antara huruf besar dan huruf kecil, sehingga memungkinkan pemrosesan teks yang lebih seragam dan akurat. Berikut ini adalah contoh penerapan *case folding* dapat dilihat pada Tabel 2.1 Contoh penerapan *case folding*.

Tabel 2.1 Contoh penerapan *case folding*

| **Sebelum** | **Sesudah** |
|---|---|
| era Habibie menurunkan dolar wajar sukses karena era Orba yg paling banyak hutang 512 T itu adalah swasta sementara BUMN sehat semua UMKM sehat era Jokowj naik suku bunga pinjaman dg semua sektor ekonomi sakit apa pengaruhnya | era habibie menurunkan dolar wajar sukses karena era orba yg paling banyak hutang 512 t itu adalah swasta sementara bumn sehat semua umkm sehat era jokowj naik suku bunga pinjaman dg semua sektor ekonomi sakit apa pengaruhnya |

### 2.3.2 *Cleansing*

*Cleansing* merupakan proses penghapusan karakter yang tidak relevan dari teks seperti tanda baca, angka, dan simbol khusus[12]. Pada penelitian ini simbol yang dihapus untuk data tweet yaitu *url*, *username*, dan *hashtag*. Berikut ini adalah contoh penerapan *cleansing* dapat dilihat pada Tabel 2.2 Contoh penerapan *cleansing*.

Tabel 2.2 Contoh penerapan *cleansing*

| **Sebelum** | **Sesudah** |
|---|---|
| era habibie menurunkan dolar wajar sukses karena era orba yg paling banyak hutang 512 t itu adalah swasta sementara bumn sehat semua umkm sehat era jokowj naik suku bunga pinjaman dg semua sektor ekonomi sakit apa pengaruhnya | era habibie menurunkan dolar wajar sukses karena era orba yg paling banyak hutang t itu adalah swasta sementara bumn sehat semua umkm sehat era jokowj naik suku bunga pinjaman dg semua sektor ekonomi sakit apa pengaruhnya |

### 2.3.3 *Tokenizing*

*Tokenizing* merupakan proses pemenggalan kalimat menjadi bagian-bagian atau kata-kata yang lebih kecil atau disebut dengan token [12]. yang disebut token [20]. Token bisa berupa kata-kata, frasa, atau simbol-simbol tertentu. Pada penelitian ini *tokenizing* dilakukan untuk memisahkan kata yang dipisahkan oleh spasi atau simbol untuk data tweet-nya. Berikut ini adalah contoh penerapan *tokenizing* dapat dilihat pada Tabel 2.3 Contoh penerapan *tokenizing*.

Tabel 2.3 Contoh penerapan *tokenizing*

| **Sebelum** | **Sesudah** |
|---|---|
| era habibie menurunkan dolar wajar sukses karena era orba yg paling banyak hutang t itu adalah swasta sementara bumn sehat semua umkm sehat era jokowj naik suku bunga | ['era', 'habibie', 'menurunkan', 'dolar', 'wajar', 'sukses', 'karena', 'era', 'orba', 'yg', 'paling', 'banyak', 'hutang', 't', 'itu', 'adalah', 'swasta', 'sementara', 'bumn', 'sehat', 'semua', 'umkm', 'sehat', 'era', |

| Sebelum | Sesudah |
|---|---|
| pinjaman dg semua sektor ekonomi sakit apa pengaruhnya | 'jokowj', 'naik', 'suku', 'bunga', 'pinjaman', 'dg', 'semua', 'sektor', 'ekonomi', 'sakit', 'apa', 'pengaruhnya'] |

### 2.3.4 *Normalization*

*Normalization* merupakan proses perbaikan kata yang disingkat dan slang yang memiliki arti serupa namun ditulis dengan cara yang berbeda [12]. Kata-kata yang diperbaiki merupakan hasil pencocokan variasi kata dari *dataset* KBBA (Kamus Besar Bahasa Indonesia) yang menginterpretasikan singkatan yang biasa digunakan dalam tweet. Berikut ini adalah contoh penerapan *Normalization* dapat dilihat pada Tabel 2.4 Contoh penerapan *Normalization*.

Tabel 2.4 Contoh penerapan *Normalization*

| Sebelum | Sesudah |
|---|---|
| ['era', 'habibie', 'menurunkan', 'dolar', 'wajar', 'sukses', 'karena', 'era', 'orba', 'yg', 'paling', 'banyak', 'hutang', 't', 'itu', 'adalah', 'swasta', 'sementara', 'bumn', 'sehat', 'semua', 'umkm', 'sehat', 'era', 'jokowj', 'naik', 'suku', 'bunga', 'pinjaman', 'dg', 'semua', 'sektor', 'ekonomi', 'sakit', 'apa', 'pengaruhnya'] | ['era', 'habibie', 'menurunkan', 'dolar', 'wajar', 'sukses', 'karena', 'era', 'orba', 'yang', 'paling', 'banyak', 'hutang', 't', 'itu', 'adalah', 'swasta', 'sementara', 'bumn', 'sehat', 'semua', 'umkm', 'sehat', 'era', 'jokowj', 'naik', 'suku', 'bunga', 'pinjaman', 'dengan', 'semua', 'sektor', 'ekonomi', 'sakit', 'apa', 'pengaruhnya'] |

### 2.3.5 *Stopword Removal*

*Stopword Removal* merupakan proses penghapusan kata yang tidak mengandung makna atau arti [12]. Kata-kata umum yang sering muncul dalam teks tetapi memiliki sedikit atau bahkan tidak ada nilai informasi saat melakukan analisis teks atau pemrosesan bahasa alami seperti “Aku”, “Dia”, dan “Ini” akan dihapus. Berikut ini adalah contoh penerapan *stopword* dapat dilihat pada Tabel 2.5 Contoh penerapan *stopword*.

Tabel 2.5 Contoh penerapan *stopword*

| **Sebelum** | **Sesudah** |
|---|---|
| ['era', 'habibie', 'menurunkan', 'dolar', 'wajar', 'sukses', 'karena', 'era', 'orba', 'yang', 'paling', 'banyak', 'hutang', 't', 'itu', 'adalah', 'swasta', 'sementara', 'bumn', 'sehat', 'semua', 'umkm', 'sehat', 'era', 'jokowj', 'naik', 'suku', 'bunga', 'pinjaman', 'dengan', 'semua', 'sektor', 'ekonomi', 'sakit', 'apa', 'pengaruhnya'] | ['era', 'habibie', 'menurunkan', 'dolar', 'wajar', 'sukses', 'era', 'orba', 'hutang', 't', 'swasta', 'bumn', 'sehat', 'umkm', 'sehat', 'era', 'jokowj', 'suku', 'bunga', 'pinjaman', 'sektor', 'ekonomi', 'sakit', 'pengaruhnya'] |

## 2.4 Topik

Topik adalah inti utama dari seluruh isi tulisan yang hendak disampaikan atau lebih dikenal dengan topik pembicaraan. Topik adalah hal yang pertama kali ditentukan ketika penulis akan membuat tulisan. Topik yang masih awal tersebut, selanjutnya dikembangkan dengan membuat cakupan yang lebih sempit atau lebih luas. Terdapat beberapa kriteria untuk sebuah topik yang dikatakan baik, di antaranya adalah topik tersebut harus mencakup keseluruhan isi tulisan, yakni mampu menjawab pertanyaan akan masalah apa yang hendak ditulis. Ciri utama dari topik adalah cakupannya atas suatu permasalahan masih bersifat umum dan tidak diuraikan secara lebih merinci [13].

Topik biasa terdiri dari beberapa kata yang singkat, dan memiliki persamaan serta perbedaan dengan tema karangan. Persamaannya adalah baik topik maupun tema keduanya dapat dijadikan sebagai judul karangan. Sedangkan, perbedaannya ialah topik masih mengandung hal yang umum, sementara tema akan lebih spesifik dan lebih terarah dalam membahas suatu permasalahan.

## 2.5 *Topic modeling*

*Topic modeling* merupakan data teks berdasarkan pengelompokan topik tertentu berdasarkan kata kunci. *Topic modeling* termasuk ke dalam *clustering*

dengan mengelompokkan dokumen berdasarkan kemiripannya [14]. Dokumen ini berasal dari hasil pengumpulan data yang sangat besar dari Twitter, yang kemudian akan digunakan dalam analisis *clustering topic modeling*.

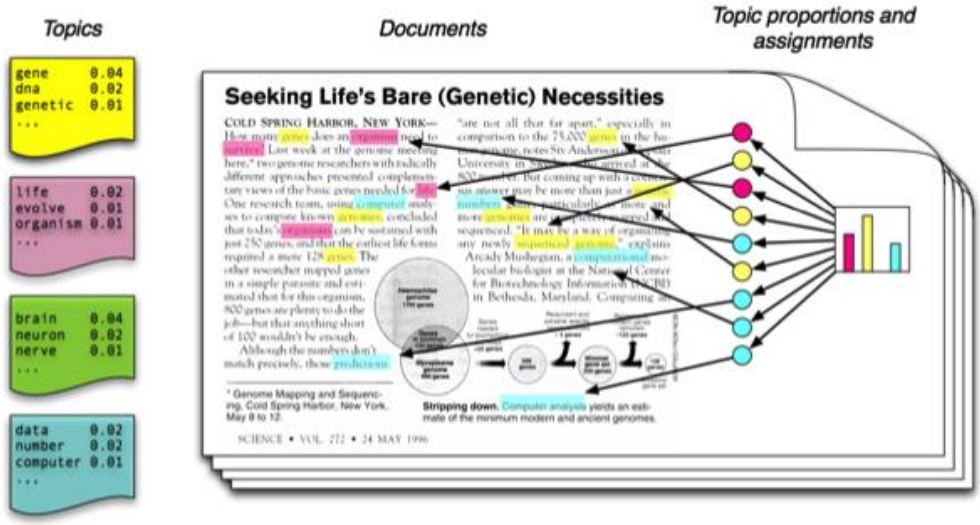



Figure source: Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, *55*(4), 77-84.

Gambar 2.1 Ilustrasi *Topic modeling*

Dalam melakukan pemodelan topik terdapat beberapa algoritma seperti *Latent dirichlet allocation* (LDA) dan *Non-negative Matrix Factorization* (NMF), namun dalam penelitian ini akan menggunakan LDA.

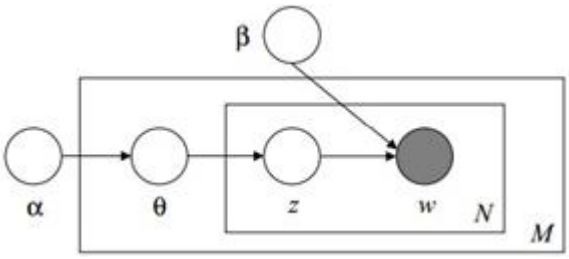
## 2.6 *Latent Dirichlet Allocation* (LDA)

*Latent Dirichlet Allocation* (LDA) adalah teknik pemodelan topik yang secara otomatis menemukan topik dalam dokumen teks. LDA menganggap dokumen sebagai campuran dari berbagai topik dan setiap kata termasuk dalam salah satu topik dokumen. LDA membayangkan serangkaian topik yang tetap, setiap topik memiliki sekumpulan kata Tujuan LDA adalah memetakan semua dokumen ke topik sedemikian rupa, sehingga kata-kata dalam setiap dokumen sebagian besar terkait dengan topik tersebut [14].

LDA adalah model probabilistik generatif dari koleksi data diskrit seperti korpus teks. Setiap topik dimodelkan sebagai campuran tak terbatas melalui set yang mendasari probabilitas topik. Dalam konteks pembuatan model teks, probabilitas topik memberikan representasi eksplisit dari sebuah dokumen. LDA

saat ini sering digunakan karena dapat melakukan *clustering*, melakukan peringkasan, menghubungkan, dan dapat memproses data dengan memberikan bobot pada masing-masing dokumen yang nantinya menghasilkan daftar topik. Ide dasar dari LDA ini menganggap bahwa dokumen yang diujikan dapat direpresentasikan sebagai sebuah model yang dicampur dari berbagai topik yang dibutuhkan [14]. oleh sebab itu teknik ini disebut sebagai laten secara formal, didefinisikan notasi sebagai berikut:

1. Kata adalah bentuk dasar dari data diskrit
2. Sebuah dokumen adalah barisan kata-kata V yang dinotasikan dengan W = ($W_1$ ,$W_2$, ...$W_n$), $w_n$ adalah barisan kata ke-n
3. Sebuah corpus adalah koleksi dari M dokumen dinotasikan dengan D = ($W_1$, $W_2$... Wn).




Gambar 2.2 *Grapical Model LDA*

Berdasarkan *graphical model* pada Gambar 2.2 *Grapical Model LDA*. Kotak-kotak tersebut adalah ”pelat” yang mewakili pengulangan. Pelat luar mewakili dokumen, sedangkan pelat dalam mewakili pilihan perulangan topik dan kata-kata dalam dokumen [14]. parameter alpha dan beta diberikan untuk corpus. Parameter alpha adalah parameter untuk distribusi topik dari dokumen, dan parameter beta adalah parameter untuk distribusi kata dari topik. Semakin besar nilai alpha, maka setiap dokumen mengandung sebagian besar topik, artinya tidak hanya ada satu topik spesifik. Sedangkan semakin besar nilai beta naka setiap topik mengandung sebagian besar kata, tidak hanya ada beberapa kata spesifik yang membedakan topik satu dengan lainnya. Untuk setiap dokumen N terdapat distribusi topik yaitu theta. Karena LDA adalah *soft clustering*, maka setiap dokumen bisa terdiri dari beberapa topik yang berbeda Kemudian menentukan topik dari setiap kata dalam setiap dokumen yang dinotasikan dengan Z, untuk nantinya kumpulkan menjadi

klaster-klaster. Maka hasilnya adalah campuran kata-kata di tiap topik/klaster yang sudah ditentukan sebelumnya kemudian diinterpretasikan hasil tiap klaster tersebut membahas topik apa [14]. Rumus perhitungan LDA sebagai berikut:

$$\boldsymbol{P(Z_t = j | z_{-t}, w_t, d_t) = \frac{c_{w.j}^{WT} + \beta}{\sum_{w=1}^{W} c_{w,j}^{WT} + W\beta} \times \frac{c_{d,j}^{DT} + \alpha}{\sum_{t=1}^{T} c_{d,t}^{DT} + T\alpha}} \quad (2.1)$$

Keterangan dari persamaan:

$P(Z_t = j)$ = Probabilitas token pada topik

$z_{-t}$ = Representasi topik dari semua token

$w_t$ = kata dari token

$d_t$ = Dokumen yang berisi token

$\beta$ = Distribusi kata per topik (parameter konsentrasi)

$W$ = Jumlah kata unik dalam dokumen

$\alpha$ = Distribusi topik per dokumen

$T$ = Jumlah Topik

$c_{w.j}^{WT}$ = Jumlah kemunculan kata pada topik

$\sum_{w=1}^{W} c_{w,j}^{WT}$ = Jumlah kemunculan topik pada matriks

$c_{d,j}^{DT}$ = Kemunculan topik dalam setiap dokumen

$\sum_{t=1}^{T} c_{d,t}^{DT}$ = Total jumlah kali setiap dokumen muncul sebagai topik

$\frac{c_{w.j}^{WT} + \beta}{\sum_{w=1}^{W} c_{w,j}^{WT} + W\beta}$ = Distribusi probabilitas kata pada suatu topik

$\frac{c_{d,j}^{DT} + \alpha}{\sum_{t=1}^{T} c_{d,t}^{DT} + T\alpha}$ = Distribusi probabilitas topik pada suatu dokumen

Setelah menghitung probabilitas dengan rumus di atas, selanjutnya dihitungan *gibbs sampling. Gibbs sampling* dilakukan untuk mendekati distribusi posterior, Selanjutnya untuk penentuan topik akhir, topik yang akan dipilih berdasarkan probabilitas tertinggi.

## 2.7 *Gibbs Sampling*

*Gibbs sampling* adalah metode yang digunakan untuk melakukan sampling dari distribusi probabilitas yang kompleks. Dalam konteks LDA, *gibbs sampling* digunakan untuk mengestimasi distribusi topik dalam dokumen dan distribusi kata dalam topik. Berikut adalah penjelasan lebih rinci tentang *gibbs sampling* dan penerapannya dalam LDA:

Ada beberapa tahap yang dilakukan untuk melakukan *gibbs sampling* dalam LDA, yaitu:

1. **Inisialisasi**: Tentukan nilai awal untuk semua variabel acak.
2. **Iterasi**:
   a. Pilih satu variabel acak.
   b. Update nilai variabel tersebut dengan sampling dari distribusi kondisionalnya, yaitu distribusi variabel tersebut dengan kondisi variabel lainnya.
   c. Ulangi langkah ini untuk semua variabel acak.
3. **Konvergensi**: Setelah beberapa iterasi, sampel yang dihasilkan akan mendekati distribusi target yang diinginkan

## 2.8 Aggregated Topic Models

Aggregated Topic Models menggabungkan beberapa model topik untuk meningkatkan koherensi topik, terutama dalam konteks seperti analisis media sosial. Prosesnya melibatkan pembuatan beberapa model topik menggunakan parameter berbeda, lalu menggabungkannya berdasarkan metrik kesamaan seperti kesamaan kosinus atau divergensi Jensen-Shannon [15]. Untuk melihat kesamaan kosinus diperlukan persamaan:

$$CS = \frac{\boldsymbol{A} \cdot \mathbf{B}}{||\boldsymbol{A}||||\boldsymbol{B}||} \quad (2.2)$$

Keterangan dari persamaan:

$CS$ = Consine Similarity

$A$ = Vektor dari topik 1

$B$ = Vektor dari topik 2

Topik dengan skor kesamaan tinggi dikumpulkan untuk membentuk model terpadu. Metode ini mengungguli model tradisional seperti LDA dan NMF dalam hal koherensi, sehingga memberikan topik yang lebih akurat dan bermakna [15].

## 2.9 *Evaluation Topic Coherence*

Dalam mengevaluasi model menggunakan Skor Koherensi atau disebut dengan *Coherence Score* atau *Topic Score* merupakan bentuk evaluasi topik yang lebih mudah dalam interpretasi oleh manusia [15]. Skor Koherensi digunakan untuk mengukur nilai suatu topik dengan melalui pengukuran tingkat kesamaan semantik dalam kata-kata yang terdapat pada topik.

Semakin tinggi nilai topik koherensi , berarti model menghasilkan kelompok-kelompok kata dalam topik yang semakin baik. Berdasarkan nilai topik koherensi, dapat pula dilakukan perhitungan untuk dapat mengurutkan topik terbaik yang dibentuk oleh model, sehingga diketahui topik mana yang benar-benar berbeda dengan topik lainnya dan memiliki kualitas yang sesuai dengan pendapat manusia.

$$PMI(w_i, w_j) = \log \frac{p(w_i, w_j)}{p(w_i)\, p(w_i)} \quad (2.3)$$

Keterangan dari persamaan:

$P(Wi)$ = Probabilitas kata pertama

$P(Wj)$ = Probabilitas kata kedua

Cara kerja dari rumus ini ada menghitung probabilitas kedua kata muncul secara bersamaan. Koherensi memiliki rentang nilai 0 – 1 [16][17], berikut ini rentang nilai standar koherensi.

Tabel 2.6 Standar koherensi

| **Rentang Nilai** | **Keterangan** |
|---|---|
| **0.5 – 1** | Baik |
| **0.3 – 0.49** | Cukup Baik |
| **0 – 0.29** | Buruk |

Pada Tabel 2.6 terlihat kapan koherensi dikatakan, baik, cukup baik, dan buruk. keterangan ini bisa digunakan sebagai standar baik atau buruknya model yang dihasilkan.

## 2.10 *Large Language Model* (LLM)

*Large Language Model* (LLM) adalah jenis model bahasa alami yang dirancang untuk memahami dan menghasilkan bahasa manusia secara alami. LLM dibangun dengan menggunakan teknik pembelajaran mesin yang disebut dengan deep learning. Model ini dilatih pada sejumlah besar teks bahasa manusia yang telah dikumpulkan dari berbagai sumber seperti korpus teks, buku, artikel, dan website [18].

LLM dapat melakukan berbagai tugas, seperti pemrosesan bahasa alami, penerjemahan mesin, pengenalan suara, dan lain-lain. Salah satu keunggulan utama LLM adalah kemampuannya untuk mempelajari pola-pola kompleks dalam bahasa manusia dan menghasilkan output yang mirip dengan cara manusia berbicara atau menulis [18]. Contoh LLM yang populer adalah GPT-3 (Generative Pre-trained Transformer 3) yang dikembangkan oleh OpenAI. LLM ini memiliki lebih dari 175 miliar parameter dan mampu melakukan berbagai tugas bahasa manusia dengan kualitas yang cukup tinggi.

### 2.10.1 Twitter

Twitter adalah layanan mikroblog yang didirikan pada awal tahun 2006 untuk memungkinkan orang berbagi pesan singkat pesan tekstual - "tweet" - dengan orang lain di dalam lain di dalam sistem [19]. Karena sistem ini awalnya dirancang untuk tweet yang akan dibagikan melalui SMS, panjang maksimum dari sebuah tweet

adalah 280 karakter. Meskipun layanan ini berevolusi untuk menyertakan lebih banyak penggunaan selain SMS, seperti klien web dan desktop.

### 2.10.2 Tweet

Tweet adalah Sebuah post (maksimal 280 karakter) dapat berisi foto, GIF, video, dan teks, me-tweet atau menge-post adalah tindakan mengirim post. Post akan ditampilkan di timeline Twitter atau melekat di situs web dan blog.

### 2.10.3 Retweet

Post orang lain yang diteruskan ke pengikut disebut sebagai retweet atau repost. Fitur ini sering digunakan untuk meneruskan berita atau temuan yang bermanfaat di Twitter, dan fitur ini selalu mempertahankan atribut aslinya. Me-retweet adalah Tindakan menyebarkan post akun lain ke semua pengikut Anda dengan mengeklik atau menyentuh tombol repost.

### 2.10.4 Follower

Follower dihasilkan dari pengguna yang mengikuti(following) akun Twitter. Pengguna Twitter dapat mengetahui jumlah follower atau pengikut yang dimiliki dari profil Twitter.

### 2.10.5 Mention

Tanda @ atau mention digunakan untuk memanggil nama pengguna dalam post: "Halo @X!" Orang lain akan menggunakan @namapengguna untuk menyebut pengguna Twitter lain di post dan mengirim pesan atau tautan ke profilnya.

## 2.11 *Crawling*

*Crawling* adalah proses otomatis yang di mana program atau bot yang dikenal sebagai web *crawler* atau *spider* mengunjungi dan mengindeksi konten dari berbagai situs web di internet. Tujuan utamanya adalah untuk mengumpulkan data dari situs web tersebut agar dapat di indeks oleh mesin pencari, sehingga informasi tersebut mudah dicari dan ditemukan oleh pengguna. *Crawler* mengunjungi halaman web, membaca konten, dan mengikuti tautan untuk menemukan halaman baru. Informasi yang dikumpulkan disimpan dalam indeks mesin pencari, yang kemudian dianalisis untuk menentukan hasil pencarian yang relevan bagi pengguna. Situs web dapat mengontrol akses *crawler* menggunakan file robots.txt,

yang memberikan instruksi tentang halaman mana yang boleh dan tidak boleh di indeks [18].

Dalam penelitian ini, *Crawling* data dilakukan untuk mengambil data dari keyword Twitter dengan bantuan pustaka Tweet Harvest di mana data tersebut dibutuhkan untuk melakukan *clustering* pada model yang akan dibangun. Tweet Harvest dibangun dengan menggunakan javascrips. Pustaka ini dapat melakukan *crawling* data tweet dengan cara memasukkan Auth Token dan kata kunci.

## 2.12 Python

Python merupakan bahasa pemrograman komputer yang biasa dipakai untuk membangun situs, *software* atau aplikasi, mengotomatiskan tugas dan melakukan analisis data. Bahasa pemrograman ini termasuk bahasa tujuan umum. Artinya, ia bisa digunakan untuk membuat berbagai program berbeda, bukan khusus untuk masalah tertentu saja. Karena sifatnya yang serba guna dan mudah digunakan, ia menjadi bahasa pemrograman yang paling banyak digunakan. Terutama untuk mereka yang masih pemula [20].

Python biasa dipakai dalam pengembangan situs dan perangkat lunak, membuat analisis data, visualisasi data dan otomatisasi tugas. Karena sifatnya yang relatif mudah dipelajari, bahasa pemrograman ini digunakan secara luas oleh non-programmer seperti ilmuwan dan akuntan untuk melakukan tugas harian mereka. Misalnya, dalam mengatur keuangan.

Python telah menjadi andalan dalam ilmu data. Bahasa pemrograman ini memungkinkan analisis data untuk melakukan perhitungan statistik yang rumit, membuat visualisasi data serta algoritma *machine learning*. Ia juga bisa digunakan untuk memanipulasi, menganalisis data, dan menyelesaikan berbagai tugas lain terkait data. Selain itu, ia bisa membantu membangun berbagai visualisasi data yang berbeda. Misalnya, grafik garis dan batang, diagram lingkaran, histogram, dan lain sebagainya. Ada beberapa fungsi dan library pada python yang dapat digunakan untuk penelitian ini, antara lain:

### 2.12.1 *Regex (Regular Expresion)*

*Regex*, atau *Regular Expressions*, adalah serangkaian karakter yang membentuk pola pencarian. Ini adalah alat yang untuk pencocokan, pengambilan, dan manipulasi teks dalam berbagai bahasa pemrograman. *Regex* sering digunakan untuk tugas-tugas seperti validasi *input*, pencarian dan penggantian teks, serta ekstraksi data. Sebagai contoh Titik(.) dalam *regex* digunakan untuk mencocokkan karakter tunggal, anda Kurung Kotak ([]) digunakan untuk mencocokkan salah satu karakter dalam set, Pangkat (^) digunakan untuk menunjukkan awal *string*, dan banyak lagi fungsi lainnya.

### 2.12.2 Gensim

Gensim adalah pustaka open-source untuk pemodelan topik dan pengolahan bahasa alami (*Natural Language Processing* atau NLP) dalam Python. Ini dirancang untuk menangani teks mentah dan menyediakan berbagai alat untuk melakukan analisis teks, pemodelan topik, vektorisasi dokumen, serta tugas-tugas NLP lainnya.

Beberapa fitur utama dalam gensim antara lain adalah Latent Semantic Analysis (LSA) dan *Latent dirichlet allocation* (LDA): Teknik untuk pemodelan topik yang mengidentifikasi pola dalam teks dan mengelompokkan dokumen berdasarkan topik. Word2Vec: Algoritma pembelajaran mendalam untuk menghasilkan representasi vektor dari kata-kata yang menangkap makna semantik. Dan lainnya

### 2.12.3 PyLDAvis

PyLDAvis adalah pustaka Python interaktif untuk visualisasi model pemodelan topik yang menggunakan LDA. Ini dirancang untuk membantu pengguna dalam menafsirkan dan memahami model LDA dengan cara yang intuitif dan mudah dipahami. PyLDAvis memungkinkan visualisasi distribusi topik, hubungan antara topik, dan distribusi kata dalam topik. Fungsi dari pustaka ini adalah menunjukkan kata-kata yang paling penting dalam setiap topik dan frekuensinya, Menunjukkan proporsi setiap topik dalam keseluruhan korpus.