

## **BAB 2**

### **TINJAUAN PUSTAKA**

#### **2.1. Blog**

Blog merupakan singkatan dari *web log* adalah bentuk aplikasi web yang berbentuk tulisan-tulisan (yang dimuat sebagai posting) pada sebuah halaman web. Tulisan-tulisan ini seringkali dimuat dalam urutan terbalik (isi terbaru dahulu sebelum diikuti isi yang lebih lama), meskipun tidak selamanya demikian. Situs *web* seperti ini biasanya dapat diakses oleh semua pengguna internet sesuai dengan topik dan tujuan dari si pengguna blog tersebut.

Media blog pertama kali dipopulerkan oleh *blogger.com*, yang dimiliki oleh Pyra Labs sebelum akhirnya diakuisisi oleh Google pada akhir tahun 2002. Semenjak itu, banyak terdapat aplikasi-aplikasi yang bersifat *open source* yang diperuntukkan kepada perkembangan para penulis blog tersebut.

Blog mempunyai fungsi yang sangat beragam, mulai dari sebuah catatan harian, media publikasi dalam sebuah kampanye politik, sampai dengan program-program media dan perusahaan-perusahaan. Sebagian blog dipelihara oleh seorang penulis tunggal, sementara sebagian lainnya oleh beberapa penulis. Banyak juga blog yang memiliki fasilitas interaksi dengan para pengunjungnya, seperti menggunakan buku tamu dan kolom komentar yang dapat memperkenankan pengunjungnya untuk meninggalkan komentar atas isi dari tulisan yang dipublikasikan, namun demikian ada juga blog yang bersifat sebaliknya (non-interaktif) [9].

#### **2.2. Spam**

Spam adalah penggunaan perangkat elektronik untuk mengirimkan pesan secara bertubi-tubi tanpa dikehendaki oleh penerimanya. Orang yang melakukan spam disebut *spammer*. Tindakan spam dikenal dengan nama *spamming* [10].

Bentuk spam yang dikenal secara umum meliputi: spam surat elektronik, spam pesan instan, spam *Usenet newsgroup*, spam mesin pencari informasi web (*web search engine spam*), spam blog, spam wiki, spam iklan baris daring, spam jejaring sosial.

Beberapa contoh lain dari spam, yaitu surel berisi iklan, surat masa singkat (SMS) pada telepon genggam, berita dalam suatu forum kelompok warta berisi promosi barang yang tidak terkait dengan kegiatan kelompok warta tersebut, *spamdexing* yang menguasai suatu mesin pencari (*search engine*) untuk mencari popularitas bagi suatu URL tertentu, berita yang tak berguna dan masuk dalam blog, buku tamu situs web dan spam transmisi faks.

Spam dikirimkan oleh pengiklan dengan biaya operasional yang sangat rendah, karena spam tidak memerlukan senarai (*mailing list*) untuk mencapai pelanggan-pelanggan yang diinginkan. Karena hambatan masuk yang rendah, maka banyak *spammers* yang muncul dan jumlah pesan yang tidak diminta menjadi sangat tinggi. Akibatnya, banyak pihak yang dirugikan. Selain pengguna Internet itu sendiri, ISP (Penyelenggara Jasa Internet atau *Internet Service Provider*), dan masyarakat umum juga merasa tidak nyaman. Spam sering mengganggu dan terkadang menipu penerimanya. Berita spam termasuk dalam kegiatan melanggar hukum dan merupakan perbuatan pidana yang bisa ditindak melalui undang-undang internet.

### **2.2.1. Spam di Blog**

*Spam* di *blog* (juga disebut hanya spam *blog*, spam komentar, atau spam sosial) adalah bentuk *spamdexing*. Hal ini dilakukan dengan posting (biasanya secara otomatis) komentar acak, menyalin materi dari tempat lain yang tidak asli, atau mempromosikan layanan komersial ke *blog*, wiki, *guestbook*, atau forum diskusi online lainnya yang dapat diakses publik. Setiap aplikasi web yang menerima dan menampilkan *hyperlink* yang dikirimkan oleh pengunjung mungkin menjadi target.

Menambahkan tautan yang mengarah ke situs web *spammer* secara artifisial meningkatkan peringkat situs di mesin pencari dimana popularitas URL berkontribusi terhadap nilai tersiratnya, contohnya adalah algoritme PageRank seperti yang digunakan oleh Google penelusuran. Hal tersebut akan meningkatkan situs komersial *spammer* yang terdaftar di depan situs lain untuk penelusuran tertentu, meningkatkan jumlah calon pengunjung dan pelanggan yang membayar.

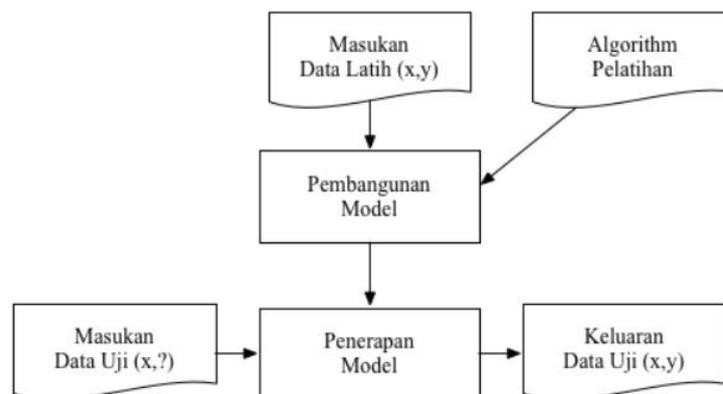
### 2.3. Klasifikasi

Klasifikasi merupakan suatu pekerjaan menilai objek data untuk memasukkannya ke dalam kelas tertentu dari sejumlah kelas yang tersedia. Dalam klasifikasi ada dua pekerjaan utama yang dilakukan. Pertama, pembangunan model sebagai *prototype* untuk disimpan sebagai memori. Kedua, penggunaan model untuk melakukan pengenalan/klasifikasi/prediksi pada suatu objek lain, agar diketahui di kelas mana objek data tersebut dalam model yang sudah disimpannya [11].

#### 2.3.1 Model

Klasifikasi dapat didefinisikan sebagai pekerjaan yang melakukan pelatihan/pembelajaran terhadap fungsi target  $f$  yang memetakan setiap set atribut (fitur)  $x$  ke satu dari sejumlah label kelas  $y$  yang tersedia. Pekerjaan pelatihan tersebut akan menghasilkan suatu model yang kemudian disimpan sebagai memori.

Model dalam klasifikasi mempunyai arti yang sama dengan kotak hitam, dimana ada suatu model yang menerima masukan, kemudian mampu melakukan pemikiran terhadap masukan tersebut, dan memberikan jawaban sebagai keluaran dari hasil pemikirannya. Kerangka kerja klasifikasi ditunjukkan pada Gambar 2.1. Pada gambar tersebut disediakan sejumlah data latih  $(x,y)$  untuk digunakan sebagai data pembangunan model. Model tersebut kemudian dipakai untuk memprediksi kelas dari data uji  $(x,y)$  sehingga diketahui kelas  $y$  yang sesungguhnya.



**Gambar 2.1 Proses Pekerjaan Klasifikasi**

Kerangka kerja seperti yang ditunjukkan pada Gambar 2.1 meliputi dua langkah proses, yaitu induksi dan deduksi. Induksi merupakan langkah untuk membangun model klasifikasi dari data latih yang diberikan, disebut juga proses pelatihan, sedangkan deduksi merupakan langkah untuk menerapkan model tersebut pada data uji sehingga kelas yang sesungguhnya dari data uji dapat diketahui.

### 2.3.2 Pengukuran Kinerja Klasifikasi

Sebuah sistem yang melakukan klasifikasi diharapkan dapat melakukan klasifikasi semua set data dengan benar, tetapi tidak dapat dipungkiri bahwa kinerja suatu sistem tidak dapat 100% benar sehingga sebuah sistem klasifikasi juga harus diukur kinerjanya. Umumnya, pengukuran kinerja klasifikasi dilakukan dengan matriks konfusi.

Matriks konfusi merupakan tabel pencatat hasil kerja klasifikasi. Tabel 2.1 merupakan contoh matriks konfusi yang melakukan klasifikasi masalah biner (dua kelas), hanya ada dua kelas, yaitu kelas 0 dan kelas 1. Setiap sel  $f_{ij}$  dalam matriks menyatakan jumlah data dari kelas  $i$  yang hasil prediksinya masuk ke kelas  $j$ . Misalnya, sel  $f_{11}$  adalah jumlah data dalam kelas 1 yang secara benar dipetakan ke kelas 1, dan  $f_{10}$  adalah data dalam kelas 1 yang dipetakan secara sah ke kelas 0.

**Tabel 2.1 Matriks Konfusi untuk Klasifikasi dua kelas**

$f_{ij}$		Kelas hasil prediksi (j)	
		Kelas = 1	Kelas = 0
Kelas asli (i)	Kelas = 1	$f_{11}$	$f_{10}$
	Kelas = 0	$f_{01}$	$f_{00}$

Dengan mengetahui jumlah data yang diklasifikasikan secara benar, kita dapat mengetahui akurasi hasil prediksi, dan dengan mengetahui jumlah data yang diklasifikasikan secara sah, kita dapat mengetahui laju eror dari prediksi yang dilakukan. Dua kuantitas ini digunakan sebagai metric kinerja klasifikasi.

Untuk menghitung akurasi digunakan formula (2.1)

$$Akurasi = \frac{\text{Jumlah data yang diprediksi secara benar}}{\text{Jumlah prediksi yang dilakukan}} = \frac{f_{11}+f_{00}}{f_{11}+f_{10}+f_{01}+f_{00}} \quad (2.1)$$

Untuk menghitung laju *error* (kesalahan prediksi) digunakan formula (2.2)

$$Laju\ error = \frac{\text{Jumlah data yang diprediksi secara salah}}{\text{Jumlah prediksi yang dilakukan}} = \frac{f_{10}+f_{01}}{f_{11}+f_{10}+f_{01}+f_{00}} \quad (2.2)$$

Semua algoritma klasifikasi berusaha membentuk model yang mempunyai akurasi tinggi atau laju eror yang rendah. Umumnya, model yang dibangun dapat memprediksi dengan benar pada semua data yang menjadi data latihnya, tetapi ketika model berhadapan dengan data uji, barulah kinerja model dari sebuah algoritma klasifikasi ditentukan.

### 2.3.3 K-Fold Cross Validation

Cross-validation adalah prosedur resampling yang digunakan untuk mengevaluasi model pembelajaran mesin pada sampel data yang terbatas.

Prosedur ini memiliki parameter tunggal yang disebut  $k$  yang mengacu pada jumlah kelompok yang sampel data yang diberikan akan dibagi menjadi. Dengan demikian, prosedur ini sering disebut *k-fold cross-validation*. Ketika nilai tertentu untuk  $k$  dipilih, itu dapat digunakan sebagai pengganti  $k$  dalam referensi ke model, seperti  $k = 10$  menjadi 10-fold cross-validation.

Validasi silang terutama digunakan dalam *machine learning* untuk memperkirakan keterampilan model *machine learning* pada data tak terlihat. Yaitu, menggunakan sampel terbatas untuk memperkirakan bagaimana model diharapkan tampil secara umum ketika digunakan untuk membuat prediksi pada data yang tidak digunakan selama pelatihan model [12].

### 2.4. Ekstraksi Fitur

Ekstraksi Fitur merupakan suatu pengambilan ciri atau fitur dari suatu bentuk yang nantinya nilai yang didapatkan akan dianalisis untuk proses selanjutnya. Pada penelitian ini akan menggunakan lima fitur yang diambil dari penelitian sebelumnya yaitu dari penelitian Bhattarai et al dan Ashwin et al yaitu: jumlah *anchor text*, ratio pengulangan kata, *post-comment similarity*, kemunculan

nama pengguna dalam kolom komentar dan selisih waktu tanggal postingan dengan komentar.

#### **2.4.1. Jumlah *Anchor Text***

Sebuah teks yang muncul dalam tag HTML `<a...>` dan `</a>` sering disebut *anchor text*. Teks ini biasanya membuat sebuah link untuk menuju sebuah halaman web yang lain dengan cara mengklik link tersebut. *Spammer* biasanya memanfaatkan hal ini dengan memasukkan *anchor text* yang mana akan menuju situs *spammer* untuk meningkatkan *page ranking* pada *search engine*. Berikut contoh komentar dengan beberapa *anchor text* dapat dilihat pada gambar 2.2.

Spy Software <http://spy-software.seo.se.com>  
Internet Spy Software <http://internet-spy-software.seo.se.com>

**Gambar 2.2 Contoh *Anchor Text***

Gambar 2.2 menunjukkan sebuah komentar yang mengandung *anchor text* yang mana dalam komentar tersebut terdapat dua *anchor text* sekaligus.

#### **2.4.2. Selisih Waktu Tanggal Postingan dengan Komentar**

Sebuah artikel biasanya ditulis karena topik tersebut lagi hangat dibicarakan banyak orang contohnya seperti topik pilkada ketika memasuki masa pilkada, atau topik piala dunia sepakbola ketika menjelang piala dunia bergulir. Komentar yang bukan spam biasanya dilakukan ketika topik tersebut masih dalam lingkup yang hangat dibicarakan sedangkan komentar spam tidak tergantung terhadap waktu. Berikut contoh sebuah artikel beserta komentarnya dapat dilihat dapat gambar 2.3, 2.4 dan 2.5.

January 17, 2005

### Blog ethics under the microscope

As much as 1960 is remembered as the first US election where television had an impact, 2004 was the 1st time CBS memos on George Bush's military record, the online pundits broke new ground. You could say that

Funnily enough, that is what two of the leading Democrat bloggers are now accused of. According to Duff Armstrong of [MyDD](#) were hired as consultants "largely in order to ensure that they said positive things

Her post was picked up by the [Wall Street Journal](#) and gained a certain currency: not least because it cost her (£130,000) from the federal government to promote the [No Child Left Behind Act](#). The [National Socialism](#) journalist but accepted he did "portray one through his commentary in print and the electronic media".

### Gambar 2.3 Contoh Sebuah Artikel

What a non story!! The only story here is O'Reilly again conflating two very different things and then announce I'm taking money, and then continue to publish my opinions--which is all that matters. The fact, pretending to be offering unbiased reports. And not in a blog either, but in the

This becomes a "story" only by thinking that if you report something about one side, you are offering in the name of "balance," to accept that in some sense the two are the same.

Comments posted by: [John Webster](#) at January 17, 2005 05:12 PM

### Gambar 2.4 Contoh Komentar Pertama dari sebuah Artikel

Get a Generic Viagra alternative at Cheap Generic ViagraGet a Cheap Generic Viagra alternative at Cheap Generic Viagra

Comments posted by: [Cheap Generic Viagra](#) at March 11, 2005 09:27 PM

### Gambar 2.5 Contoh Komentar Kedelapan dari sebuah Artikel

Dari gambar 2.3, artikel ditulis pada tanggal 17 Januari 2005, kemudian gambar 2.4 menunjukkan komentar bukan spam dilakukan pada hari yang sama, yang mana artinya komentar tersebut mungkin dilakukan beberapa jam setelah artikel tersebut dipublikasikan, sedangkan pada gambar 2.5 menunjukkan komentar spam dilakukan pada tanggal 11 Maret 2005 yang mana dilakukan dua bulan setelah artikel tersebut dipublikasi dan topik yang dibicarakan dalam artikel tersebut mungkin sudah mulai ditinggalkan dan tidak menarik lagi.

#### 2.4.3. Kemunculan Nama Pengguna dalam Kolom Komentar

Sistem komentar selalu menyediakan kolom nama yang mana digunakan untuk memasukkan nama pengkomenter. Komentar yang bukan spam secara umum tidak akan memasukkan namanya di dalam komentarnya sedangkan *spammer* menggunakan *keywords* sebagai namanya dan memasukkannya ke dalam

komentar, hal tersebut bertujuan untuk meningkatkan *keywords* pada *search engine*. Berikut contoh kemunculan nama pengguna dalam kolom komentar dapat dilihat pada gambar 2.6.

Get a Generic Viagra alternative at Cheap Generic ViagraGet a Cheap Generic Viagra alternative at Cheap Generic Viagra  
Comments posted by: Cheap Generic Viagra at March 11, 2005 09:27 PM

### **Gambar 2.6 Contoh Komentar Kemunculan Nama Pengguna dalam Kolom Komentar**

Pada gambar 2.6 seseorang berkomentar dengan nama pengguna Cheap Generic Viagra kemudian di dalam isi komentarnya, orang tersebut menyelipkan nama yang ia gunakan. Inilah yang disebut munculnya nama pengguna dalam kolom komentar.

#### **2.4.4. Ratio Pengulangan Kata**

Komentar spam biasanya menggunakan kata-kata yang berulang untuk menarik perhatian *search engine* sedangkan komentar yang bukan spam biasanya menggunakan bahasa yang mengalir dan lebih *natural*. Berikut contoh perhitungan yang akan digunakan untuk menghitung ratio pengulangan kata ditunjukkan oleh persamaan (2.3) dan contoh komentar yang mengandung pengulangan kata pada gambar 2.7.

$$\text{Ratio Pengulangan Kata} = 1 - \frac{\text{Jumlah kata unik di komentar}}{\text{Jumlah total kata di komentar}} \quad (2.3)$$

Get a Generic Viagra alternative at Cheap Generic ViagraGet a Cheap Generic Viagra alternative at Cheap Generic Viagra  
Comments posted by: Cheap Generic Viagra at March 11, 2005 09:27 PM

### **Gambar 2.7 Contoh Komentar Mengandung Pengulangan Kata**

Dari gambar 2.7 jumlah total kata yang terdapat pada komentar berjumlah 18 kata dan jumlah kata unik dalam komentarnya berjumlah 8 kata. Jika dihitung dengan persamaan (2.3) didapatkan nilai ratio pengulangan katanya adalah 0,55.

#### **2.4.5. Post-Comment Similarity**

*Spammers* biasanya menggunakan script otomatis dalam melakukan tindakan spam. Oleh karena itu, komentar spam biasanya tidak terkait dengan

konteks postingan. Berikut adalah contoh komentar spam yang tidak terkait dengan postingan dapat dilihat pada gambar 2.8.

read your blog daily good post

Posted by: [trannies](#) at March 26, 2005 11:15 PM

### **Gambar 2.8 Contoh Komentar yang Tidak Relevan**

Gambar 2.7 menunjukkan komentar spam yang tidak memiliki keterkaitan dengan konteks postingan. Tipe komentar seperti itu sangat sulit dibedakan dengan komentar bukan spam karena terlihat seolah-olah seperti komentar bukan spam. Untuk menghitung nilai *post-comment similarity* akan menggunakan modul string-similarity yang terdapat pada NPM.

## **2.5. Preprocessing**

*Preprocessing* adalah proses pengolahan data asli yang akan dipersiapkan untuk diolah pada tahap selanjutnya. Masukan data awal yaitu berupa dokumen. Tahapan preprocessing yang dilakukan pada penelitian ini meliputi *Case Folding*, *Filtering*, *Tokenization*, *Stopword Removal*, dan *Stemming*.

### **2.5.1. Case Folding**

*Case folding* adalah tahapan pemrosesan teks dimana semua teks diubah ke dalam *case* yang sama. Pada penelitian ini semua huruf dalam teks dokumen diseragamkan menjadi huruf kecil. Tahapan-tahapan yang dilakukan yaitu:

1. Membaca data masukan berupa teks.
2. Mengecek keberadaan huruf kapital (Jika menggunakan lower case). Jika ada maka ke tahap 3, jika tidak ada maka algoritma selesai.
3. Lakukan perubahan string lower pada setiap string.
4. Algoritma selesai.

### **2.5.2. Filtering**

*Filtering* mengacu pada proses memutuskan istilah mana yang harus digunakan untuk merepresentasikan dokumen sehingga dapat digunakan untuk menggambarkan isi dokumen dan membedakan dokumen dengan dokumen lain

yang ada pada koleksi. Dengan kata lain filtering merupakan proses dimana teks selain karakter “a” sampai “z” dan spasi akan dihilangkan. Tahapan-tahapan yang dilakukan yaitu :

1. Membaca data masukan berupa teks.
2. Mengecek apakah ada huruf selain “a” sampai “z” dan “ ” (spasi). Jika tidak ditemukan, maka algoritma berakhir. Jika ada maka ke tahap 3.
3. Hapus karakter yang bukan termasuk “a” sampai “z” dan “ ” (Spasi).
4. Algoritma selesai.

### 2.5.3. Tokenization

Proses *tokenization* adalah proses pemotongan string input berdasarkan tiap kata yang menyusunnya, memecahkan dokumen dan memisahkannya menjadi setiap kata. Pemecahan dokumen menjadi kata-kata tunggal dilakukan dengan *scan* dokumen dan setiap kata teridentifikasi atau terpisahkan dengan kata yang lain oleh pemisah spasi. Fitur ini terdiri dari tipe kapitalisasi (tipe case), keberadaan digit, tanda baca, karakter spesial dan lain sebagainya. Hasil keluaran dari proses *tokenization* akan dipergunakan sebagai masukan dalam tahap transformasi teks. Tahapan-tahapan yang dilakukan dalam *tokenization* yaitu:

1. Membaca data masukan berupa teks.
2. Memeriksa apakah sudah EOF (*End of File*). Jika sudah maka algoritma berakhir, jika belum maka ke tahap 3.
3. Memeriksa teks dengan memecahnya menjadi kata (*term*). Melakukan pengecekan apakah menemukan spasi. Jika bertemu spasi maka ke tahap 4. Jika tidak maka *scan* kembali.
4. Mengambil kata (*term*) dari teks dan membuat indeks baru untuk menandai asal kata (*term*).
5. Algoritma selesai.

### 2.5.4. Stopword Removal

Proses pembuangan *stopword* dimaksudkan untuk mengetahui suatu kata masuk ke dalam *stopword* atau tidak. *Stoplist* berisi kumpulan kata yang ‘tidak relevan’, tetapi seringkali muncul dalam sebuah dokumen. *Stoplist* yang akan

digunakan yaitu dari <https://www.ranks.nl/stopwords>. Dengan kata lain, *stoplist* berisi sekumpulan *stopwords*. Stopwords removal adalah proses menghilangkan kata yang ‘tidak relevan’ dari sebuah dokumen teks dengan cara membandingkannya dengan *stoplist* yang ada.

*Stopword* biasanya berisi kata-kata yang sering kali muncul berupa kata sambung, kata depan, kata ganti, kata penghubung, dll. Namun artinya tidak deskriptif dan tidak memiliki keterkaitan dengan topik tertentu. Tahapan-tahapan yang dilakukan yaitu :

1. Membaca data masukan berupa teks.
2. Melakukan pengecekan apakah ada kata yang termasuk dalam *stoplist*, jika tidak ada maka algoritma selesai, jika ada ke tahap 3.
3. Menghapus semua kata yang termasuk dalam *stoplist*.
4. Algoritma selesai.

### **2.5.5. Stemming**

*Stemming* merupakan suatu proses untuk menemukan kata dasar dari sebuah kata dengan menghilangkan semua imbuhan. Berdasarkan asumsi bahwa term yang memiliki akar kata yang sama akan selalu memiliki makna yang sama, stemming digunakan pada information retrieval untuk meningkatkan keakuratan retrieval (perolehan informasi).

Porter stemmer merupakan salah satu algoritma stemming yang telah lama digunakan. Porter Stemmer pertama kali dipublikasikan pada tahun 1980. Selain itu, Porter stemmer merupakan stemmer ini tidak menggunakan kamus, karena berdasarkan kenyataannya resource seperti kamus digital yang besar sangatlah mahal.

### **2.6. Pembobotan Fitur**

Pembobotan fitur adalah suatu pembentukan nilai vektor yang berdasarkan dari hasil ekstraksi setiap fitur yang digunakan sehingga suatu dokumen akan memiliki ciri tersendiri dengan dokumen lainnya berdasarkan fitur-fitur yang digunakan.

## 2.7. Normalisasi Vektor

Normalisasi vektor adalah membuat panjang sebuah vektor menjadi satu. Berikut rumus yang digunakan untuk menormalisasikan vektor ditunjukkan oleh persamaan (2.4):

$$V_{normal} = \frac{v}{|v|} \quad (2.4)$$

## 2.8. Metode Rocchio Classification

*Rocchio classifiers* merupakan salah satu metode pembelajaran *supervised document classification*. Metode klasifikasi rocchio membandingkan kesamaan isi antara data training dan data test dengan merepresentasikan semua data ke dalam sebuah vector. Dalam menggunakan *vector space model* diperlukan batas-batas antar kelas untuk mengetahui klasifikasi yang sesuai. Teknik Rocchio menerapkan batas-batas tersebut dalam bentuk centroid untuk memberi batasan tersebut. Centroid sebuah kelas  $c$  adalah rata-rata semua vektor yang berada pada kelas  $c$ . Untuk menghitung nilai centroid dapat dilihat pada persamaan (2.5) [13].

$$\vec{\mu}(c) = \frac{1}{|D_c|} \sum_{d \in D_c} \vec{v}(d) \quad (2.5)$$

Dimana  $D_c$  adalah himpunan dokumen di dalam korpus pada *class*  $c$  dan  $\vec{v}(d)$  adalah vektor  $d$  yang telah dinormalisasi. Untuk menentukan kemiripan dua vektor *vector space model* yaitu dengan mengukur jarak. Dalam menentukan jarak (distance) antara dua vektor space model digunakan jarak euclidean yang dapat dilihat pada persamaan (2.6) dimana  $p$  dan  $q$  adalah vektor [10].

$$\begin{aligned} d(\mathbf{q}, \mathbf{p}) &= \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} \\ &= \sqrt{\sum_{i=1}^n (q_i - p_i)^2}. \end{aligned} \quad (2.6)$$

## 2.9. Pemrograman Terstruktur

Pemrograman terstruktur adalah konsep atau paradigma atau sudut pandang pemrograman yang membagi – bagi program berdasarkan fungsi – fungsi atau prosedur – prosedur yang dibutuhkan program komputer. Modul – modul

(pembagian program) biasanya dibuat dengan mengelompokan fungsi – fungsi dan prosedur – prosedur yang diperlukan sebuah proses tertentu [14].

Fungsi – fungsi dan prosedur – prosedur ditulis secara sekuensial atau terurut dari atas ke bawah sesuai dengan kebergantungan antar fungsi atau prosedur (fungsi atau prosedur yang dapat dipakai oleh fungsi atau prosedur dibawahnya harus yang sudah ditulis atau dideklarasikan di atasnya).

## **2.10. DFD (Data Flow Diagram)**

Data Flow Diagram (DFD) awalnya dikembangkan oleh Chris Gane dan Trish Sarson pada tahun 1979 yang termasuk dalam Structured Systems Analysis and Design Methodology (SSADM) yang ditulis oleh Chris Gane dan Trish Sarson Sistem yang dikembangkan ini berbasis pada dekomposisi fungsional dari sebuah sistem.

Edward Yourdon dan Tom DeMarco memperkenalkan metode yang lain pada tahun 1980-an di mana mengubah persegi dengan sudut lengkung dengan lingkaran untuk menotasikan. DFD Edward Yourdon dan Tom DeMarco populer digunakan sebagai model analisis sistem perangkat lunak untuk sistem perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.

DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. DFD dapat dibagi menjadi beberapa level yang lebih detail untuk merepresentasikan aliran informasi atau fungsi yang lebih detail.

## **2.11. Software Aplikasi**

Dalam pembangunan sistem klasifikasi komentar pada penelitian ini dibutuhkan *software-software* aplikasi sebagai sarana terwujudnya sistem yang akan dibangun.

### **2.11.1. VueJS**

VueJs adalah javascript framework yang digunakan untuk membangun *user interface*. VueJs merupakan framework javascript yang sangat populer yang mana dapat memudahkan dalam pengorganisasian dalam pembangunan web yang dibuat

secara sederhana. Vue memiliki fitur-fitur yang menarik dalam rangka memudahkan penggunaannya dengan menyediakan tools seperti Vuex untuk *state management* dan Vue Router untuk pembangunan rute aplikasi. [15].