# BAB 2
# LANDASAN TEORI

## 2.1. Peringkas Teks Otomatis

Dalam Kamus Besar Bahasa Indonesia, ringkasan merupakan ikhtisar, singkatan cerita, meringkaskan dan kependekan. Sedangkan kata otomatis yaitu secara otomat, bekerja sendiri dan sendirinya. Ringkasan didefinisikan sebagai penyajian karangan atau peristiwa yang panjang dalam bentuk yang singkat dan efektif, ringkasan ini dapat mempertahankan urutan isi ataupun sudut pandang atau gagasan dengan mengambil inti sari dari keseluruhan teks.

Peringkas teks otomatis merupakan proses pemampatan teks sumber kedalam versi yang lebih pendek, namun tetap mempertahankan informasi yang terkandung didalamnya [**7**]**.**

### 2.1.1 Bentuk Ringkasan

Menurut Zaman B. dan E Winarko terdapat 2 pendekatan peringkas teks [**8**], yaitu:

1. **Ringkasan dalam bentuk Abstrak**

   Teknik abstraksi menggunakan metode linguistik untuk memeriksa dan menafsirkan teks dokumen menjadi ringkasan. Ringkasan teks tersebut dihasilkan dengan cara menambahkan kalimat-kalimat baru yang merepresentasikan intisari teks sumber ke dalam bentuk yang berbeda dengan kalimat-kalimat yang ada pada teks sumber. Pada penelitian ini metode pendekatan yang digunakan untuk peringkasan teks otomatis yaitu teknik ekstraktif dengan memanfaatkan aplikasi yang dijalankan pada komputer untuk menghasilkan sebuah ringkasan dari dokumen aslinya.

**2. Ringkasan dalam bentuk Ekstraksi**

Ekstraksi (*extractive summary*) Pada teknik ekstraksi, sistem menyalin unit-unit teks yang dianggap paling penting dari teks sumber menjadi ringkasan. Unit-unit teks yang disalin dapat berupa klausa utama, kalimat utama, atau paragraf utama tanpa ada penambahkan kalimat-kalimat baru yang tidak terdapat pada dokumen aslinya.

## 2.2. *Preprocessing*

Pada tahap ini dilakukannya proses *preprocessing* dimana langkah ini merupakan tahapan awal dalam mengolah data input sebelum memasuki tahapan utama. Tujuan dari tahap ini adalah untuk penyeragaman dan untuk memudahkan dalam membaca. *Preprocessing* sendiri terdiri atas : proses pemisahan kalimat, *Case Folding*, *filtering*, *tokenizing*, *stopword removal* dan proses *stemming* [**9**].

**1. Pemisahan Kalimat**

Langkah awal dalam proses *preprocessing* adalah memisahkan kalimat. Proses ini dilakukan untuk memisahkan kalimat didalam sebuah teks, sehingga didapatkannya sekumpulan kalimat. Pada tahapan ini pemisahan kalimat dilakukan jika menemukan tanda titik (. ), tanda tanya (? ) dan tanda seru (! ).

**2. *Case Folding***

Tahapan ini merupakan tahapan lanjutan dari pemisahan kalimat. Tahapan ini akan mengubah seluruh huruf besar menjadi huruf kecil.

**3. *Filltering***

Teks yang sudah dipisahkan menjadi kumpulan kalimat dan dikecilkan semua hurufnya, akan dilanjutkan dengan proses *filltering* dimana proses ini akan menghilangkan semua karakter selain karakter ‘a’ sampai dengan ‘z’ dan hanya menerima spasi serta angka (0-9)..

**4. *Tokenizing***

Proses *tokenizing* ini merupakan proses pemotongan string input berdasarkan tiap kata yang menyusunnya. Pemisahan kalimat menjadi kata – kata tunggal dilakukan dengan cara melakukan pencarian kalimat, dimana kalimat yang memiliki *space* akan dijadikan sebuah kata – kata tunggal.

**5. *Stopword Removal***

*Stopword removal* ini merupakan proses menghilangkan kata – kata yang sering muncul, namun kata – kata tersebut tidak memiliki arti. Dikatakan tidak berarti jika tidak memiliki keterkaitan dengan topik. Untuk mengetahui kata apa saja yang dikategorikan *stopword* maka digunakan kamus *stopword* [**10**]. Contoh *stopword* dalam bahasa indonesia adalah ini, itu , ke, dari, pada, dan lain –lain.

**6. *Stemming***

*Stemming* merupakan proses mencari akar (*root*) kata dari tiap token kata, yaitu dengan pengembalian suatu kata berimbuhan ke bentuk dasarnya (*stem*) [**11**]. Pada bahasa Indonesia, proses ini dilakukan dengan menghilangkan imbuhan yang mengawali dan mengakhiri kata sehingga diperoleh bentuk dasar dari kata tersebut. Sebagai contoh, kata berpergian, kepergian akan di *stem* menjadi pergi. Algoritma yang digunakan dalam proses stemming ini merupakan algoritma *Porter Stemmer* bahasa Indonesia.

Algoritma ini dipakai karena cocok dengan strukur morfologi kata – kata di dalam bahasa Indonesia. Algoritma ini sebenarnya mengambil dari algoritma *English Porter Stemmer*. Proses ini bisa dilihat pada Gambar 2. 1




**Gambar 2. 1 Gambaran Algoritma *Porter Stemmer* Bahasa Indonesia**

Algoritma ini memiliki 5 aturan pada proses *stemming* untuk bahasa Indonesia menggunakan *porter stemmer*, dengan 5 aturan sebagai berikut :

a. Penanganan terhadap partikel infleksional, yaitu : lah, tah, kah. Contoh : duduklah, makanlah. Contoh pembentukan partikel dapat dilihat pada tabel 2.1.
b. Penanganan terhadap kata ganti infleksional, yaitu ku, mu, nya. Contoh : sepedaku, bukunya. Contoh pembentukan kata ganti infleksional dapat dilihat pada tabel 2.2.
c. Penanganan terhadap prefiks derivasional pertama, yaitu : meng dan semua variasinya, peng dan semua variasinya, di ter, dan ke. Contoh :

membakar, pengukur, kekasih. Contoh penanganan terhadap prefiks derivasional pertama dapat dilihat pada tabel 2.3.

d. Penangannan terhadap prefix derivasional kedua, yaitu : ber dan semua variasinya serta per dan semua variasinya. Contoh : berlari, belajar, perkata. Contoh penanganan terhadap prefix derivasional kedua dapat dilihat pada tabel 2.4.
e. Penangannan terhadap Sufiks derivasional, yaitu : kan, an dan i. Contoh : ambilkan, janjian dan dapati. Contoh penanganan terhadap surfiks derivasional dapat dilihat pada tabel 2.5.

Berdasarkan apa yang telah diuraikan, terdapat 5 kelompok aturan yang dapat dilihat pada tabel – tabel dibawah ini.

Pada Tabel 2. 1 menjelaskan tentang pembentukan partikel.

**Tabel 2. 1 Pembentukan Partikel**

| Akhiran | Pengganti | Kondisi Ukuran | Kondisi Tambahan | Contoh |
|---|---|---|---|---|
| **-kah** | NULL | 2 | NULL | Sapukah → sapu |
| **-lah** | NULL | 2 | NULL | Datanglah → datang |
| **-pun** | NULL | 2 | NULL | Bukupun → buku |

Pada Tabel 2. 2 menjelaskan tentang pembentukan kata ganti.

**Tabel 2. 2 Pembentukan Kata Ganti**

| Akhiran | Pengganti | Kondisi Ukuran | Kondisi Tambahan | Contoh |
|---|---|---|---|---|
| -ku | NULL | 2 | NULL | bukuku → buku |
| -mu | NULL | 2 | NULL | bukumu → buku |
| -nya | NULL | 2 | NULL | bukunya → buku |

Pada Tabel 2. 3 menjelaskan tentang pembentukan awalan pertama.

**Tabel 2. 3 Pembentukan Awalan Pertama**

| Awalan | Pengganti | Kondisi Ukuran | Kondisi Tambahan | Contoh |
|---|---|---|---|---|
| meng- | NULL | 2 | NULL | mengukur → ukur |
| meny- | s | 2 | V...* | menyapu → sapu |

| Awalan | Pengganti | Kondisi Ukuran | Kondisi Tambahan | Contoh |
|---|---|---|---|---|
| men- | NULL | 2 | NULL | menduga → duga<br>menuduh → uduh |
| mem- | p | 2 | V... | memilah → pilah |
| mem- | NULL | 2 | NULL | membaca → baca |
| me- | NULL | 2 | NULL | merusak → rusak |
| peng- | NULL | 2 | NULL | pengukur – ukur |
| peny- | S | 2 | V... | penyapu → sapu |
| pen- | NULL | 2 | NULL | penduga → duga<br>penuduh → uduh |
| pem- | P | 2 | V.. | pemilah → pilah |
| pem- | NULL | 2 | NULL | pembaca → baca |
| di- | NULL | 2 | NULL | diukur → ukur |
| ter- | NULL | 2 | NULL | tersapu → sapu |
| Ke- | NULL | 2 | NULL | kekasih → kasih |

Pada Tabel 2. 4 menjelaskan tentang pembentukan awalan kedua.

**Tabel 2. 4 Pembentukan Awalan Kedua**

| Awalan | Pengganti | Kondisi Ukuran | Kondisi Tambahan | Contoh |
|---|---|---|---|---|
| ber- | NULL | 2 | NULL | berlari → lari |
| bel- | NULL | 2 | Ajar | belajar → ajar |
| be- | NULL | 2 | Kerja | bekerja → kerja |
| per- | NULL | 2 | NULL | perjelas → jelas |
| pel- | NULL | 2 | Ajar | pelajar → ajar |
| pe- | NULL | 2 | NULL | pekerja → kerja |

Tabel 2. 5 menjelaskan tentang pembentukan Akhiran.

**Tabel 2. 5 Pembentukan Akhiran**

| Awalan | Pengganti | Kondisi Ukuran | Kondisi Tambahan | Contoh |
|---|---|---|---|---|
| -kan | NULL | 2 | Awalan bukan anggota {ke, | berlari → lari |

| Awalan | Pengganti | Kondisi Ukuran | Kondisi Tambahan | Contoh |
|---|---|---|---|---|
| | | | peng} | |
| -an | NULL | 2 | Awalan bukan anggota {di, meng, ter} | belajar → ajar |
| -i | NULL | 2 | Awalan bukan Anggota {ber, ke, peng} | bekerja → kerja |

Pada algoritma porter pada bahasa Indonesia ini memiliki masalah karena struktur morfologi dalam bahasa indonesia ini memiliki tingkat kerumitan yang lebih tinggi dari pada bahasa Inggris. Misalnya, sulit membedakan kata yang mengandung imbuhan baik awalan maupun akhiran dengan kata dasar yang memiliki bagian dari imbuhan, contoh :

1. Duduklah → duduk
   dilakukan proses *stemming* karena terdapat imbuhan akhiran – lah.
2. Sekolah → seko
   Dilakukan proses stemming karena terdapat imbuhan akhiran – lah, seharusnya ini tidak dilakukan proses stemming karena sekolah merupakan kata kerja.

Dari contoh diatas dapat disimpulkan, proses stemming pada algoritma porter memiliki kelemahan pada proses *stemming* kata dasar yang menyerupai kata imbuhan.

Untuk mengatasi itu semua, perlu ditambahkannya sebuah kamus [**10**] kata kerja untuk mempermudah proses *stem* agar mendapatkan hasil yang akurat.

## 2.3 Pembobotan Fitur Kalimat

Pada tahap ini dilakukan pemilihan kalimat sesuai fitur yang ada. Fitur tersebut terdiri dari [**5**] :

1. Frekuensi Kalimat

Fitur ini bisa dilakukan setelah penghilangan kata pada *stopword* dan penghilangan imbuhan pada teks, dimana secara umum metode ini menggunakan metode tf-idf dalam menghitung frekuensi kata.

$$wi = tfi * log\frac{\mathrm{N}}{\mathrm{n}} \quad (\mathbf{2.1})$$

Dimana :

- tfi : Banyaknya kata yang dicari dalam sebuah dokumen
- N : Total Dokumen / Kalimat
- n : Banyak dokumen yang mengandung kata yang dicari

$$F1 = \frac{\sum_{i=1}^{k} Wi}{Max\ (Wi)} \ (\mathbf{2.2})$$

Dimana :

- k : jumlah kata dalam kalimat

2. Kata kunci dalam kalimat

Pada Fitur kedua ini Kata kunci didapat dari kata yang muncul secara berulang kali, dimana setiap kata yang termasuk kedalam kata kunci akan diberi nilai 0,1. Nilai 0,1 didapat dari penelitian sebelumnya [**12**]. Lalu jumlahkan kata kunci disetiap kalimat. Pada fitur kata kunci akan ada kondisi, dimana kata kunci dalam kalimat tidak ditemukan dan menghasilkan nilai 0. Karena dengan adanya nilai 0 akan bermasalah pada proses selanjutnya. Oleh karena itu, fitur kata kunci akan dijumlahkan dengan nilai 0,1 disetiap kalimatnya.

3. Kemiripan Kata terhadap Judul

Fitur ketiga ini akan menghitung kata mana saja yang dimiliki suatu kalimat yang relevan dengan judul dokumen.

$$\text{F3} = \frac{Kata\ dalam(i) \cap Kata\ dalam\ judul}{Kata\ dalam(i) \cup Kata\ dalam\ judul} + 0{,}1 \quad \textbf{(2.3)}$$

Pada fitur kemiripan kata terhadap judul juga akan ada kondisi dimana kalimat tidak memiliki kata yang sama dengan judul dan menyebabkan nilai 0. Oleh karena itu, fitur ini akan dijumlahkan dengan 0,1.

4. Kemiripan antar kalimat

   Kemiripan antar-kalimat merupakan kata yang muncul dalam kalimat sama dengan kata yang muncul dalam kalimat lain [**13**].

$$\text{F4} = \frac{Kata\ dalam(i) \cap Kata\ antar\ kalimat}{Kata\ dalam(i) \cup Kata\ antar\ kalimat} \textbf{(2.4)}$$

   Dimana i : posisi kalimat

5. Lokasi Kalimat

   Fitur ini didasarkan pada asumsi bahwa kalimat pertama dan kalimat terakhir pada paragraf merupakan bagian paling penting. Hasil ini didapatkan dari.

$$F5 = Max\left(\frac{1}{i}, \frac{1}{n-i+1}\right) \quad (\mathbf{2.5})$$

   Dimana i : posisi kalimat

   n : jumlah kalimat

6. Panjang Kalimat

   Fitur ini didasarkan pada asumsi bahwa kalimat terpanjang biasanya memiliki banyak bagian paling penting

$$F6 = \frac{panjang\ kalimat\ (i)}{kalimat\ terpanjang\ dalam\ dokumen} (\mathbf{2.6})$$

   Dimana i : posisi kalimat

## 2.4 *Analytical Hierarchy Process* (AHP)

Proses pengambilan keputusan pada dasarnya adalah memilih suatu alternatif. Pada dasarnya AHP ini menggunakan persepsi manusia untuk membuat keputusan. Dengan persepsi tersebut, suatu masalah kompleks dan

tidak terstruktur dipecahkan ke dalam kelompok-kelompoknya. Kemudian kelompok-kelompok tersebut diatur menjadi suatu bentuk hirarki [**14**].

AHP adalah suatu perangkat untuk pengambilan keputusan. Bentuknya sederhana, fleksibel dan berdaya guna besar (*powerfull*) untuk mendukung suatu proses pengambilan keputusan dengan multi kriteria, multi tujuan dan penuh dengan situasi kompleks. Perangkat ini juga seringkali digunakan untuk menentukan pilihan dari berbagai alternatif yang sulit [**14**].

AHP banyak digunakan karena AHP dapat mengembangkan kemampuan seseorang untuk menggunakan logikanya dalam menghadapi permasalahan yang kompleks dan rumit. Hal ini dimungkinkan karena AHP menyediakan suatu prosedur untuk membuat prioritas terhadap pilihan-pilihan yang kaku, baik itu alternatif tindakan, perencanaan ataupun kebijakan [**14**].

AHP bekerja berdasarkan kombinasi input berbagi pertimbangan dari pembuat keputusan yang didasarkan pada informasi tentang elemen-elemen pendukung keputusan tersebut, yaitu untuk menentukan suatu set pengukuran prioritas dalam rangka evaluasi terhadap berbagai alternatif yang akan diambil dalam suatu produk keputusan [**14**].

## A. Konsep Dasar AHP

Prosedur AHP dimulai dengan dengan identifikasi berbagai elemen pendukung keputusan dan melakukan penilaian atasnya berdasarkan tingkat kepentingan, preferensi atau keberpihakan. Elemen-elemen ini dapat berupa alternatif tindakan, kriteria dan atribut yang pada akhirnya akan digunakan untuk menentukan prioritas atau peringkat dari serangkaian alternatif keputusan yang akan diambil [**14**]. Konsep dasar AHP adalah :

1. Penyusunan Hierarki

   Penyusunan hirarki permasalahan merupakan langkah untuk mendefinisikan masalah yang rumit dan kompleks, sehingga menjadi jelas dan rinci. Keputusan yang akan diambil ditetapkan sebagai tujuan, yang dijabarkan menjadi elemen-elemen yang lebih rinci hingga mencapai suatu tahapan yang paling operasional/terukur.

Hirarki tersebut memudahkan pengambil keputusan untuk memvisualisasikan permasalahan dan faktor-faktor terkendali dari permasalahan tersebut. Hirarki keputusan disusun berdasarkan pandangan dari pihak-pihak yang memiliki keahlian dan pengetahuan di bidang yang bersangkutan [**14**].

2. Penilaian Kriteria dan Alternatif

Untuk membuat *pairwise comparison*, Saaty membuat skala perbandingan yang disebutnya sebagai **skala fundamental** yang diturunkan berdasarkan riset psikologis atas kemampuan individu dalam membuat suatu perbandingan secara berpasangan terhadap beberapa elemen yang akan diperbandingkan. Skala perbandingan terebut adalah sebagai berikut :

**Tabel 2. 6 Skala Perbandingan**

| Intensitas Kepentingan | Definisi |
|---|---|
| 1 | Kedua elemen **sama pentingnya** |
| 3 | Salah satu elemen **sedikit lebih penting** |
| 5 | Salah satu elemen **jelas lebih penting** |
| 7 | Salah satu elemen **sangat jelas lebih penting** |
| 9 | Salah satu elemen **paling penting** |
| 2.4.6.8 | Apabila ragu-ragu antara dua nilai yang berdekatan |

**3.** Penentuan Prioritas

Untuk setiap kriteria dan alternatif, perlu dilakukan perbandingan berpasangan (*pairwise comparisons*). Nilai-nilai perbandingan relatif kemudian diolah untuk menentukan peringkat relatif dari seluruh alternatif. Baik kriteria kualitatif, maupun kriteria kuantitatif, dapat dibandingkan sesuai dengan *judgment* yang telah ditentukan untuk menghasilkan bobot dan prioritas dihitung dengan manipulasi matriks atau melalui penyelesaian matematik.

**4.** konsistensi Logis

Semua elemen dikelompokkan secara logis dan diperingkatkan secara konsisten sesuai dengan suatu kriteria yang logis.

Kelebihan AHP dibandingkan dengan yang lainnya adalah :

1. Struktur yang berhirarki, sebagai konsekuensi dari kriteria yang dipilih, sampai pada subkriteria yang paling dalam.
2. Memperhitungkan validitas sampai dengan batas toleransi inkonsistensi berbagai kriteria dan alternatif yang dipilih oleh para pengambil keputusan
3. Memperhitungkan daya tahan atau ketahanan output analisis sensitivitas pengambilan keputusan.

## B. Prosedur AHP

Secara umum langkah-langkah yang harus dilakukan dalam menggunakan AHP untuk pemecahan suatu masalah adalah sebagai berikut:

1. Mendefinisikan masalah dan menentukan solusi yang diinginkan, lalu menyusun hierarki dari permasalahan yang dihadapi.
2. Menentukan prioritas elemen
   a. Langkah pertama dalam menentukan prioritas elemen adalah membuat perbandingan pasangan, yaitu membandingkan elemen secara berpasangan sesuai kriteria yang diberikan.
   b. Matriks perbandingan berpasangan diisi menggunakan bilangan untuk merepresentasikan kepentingan relatif dari suatu elemen terhadap elemen yang lainnya.
3. Pertimbangan-pertimbangan terhadap perbandingan berpasangan disintesis untuk memperoleh keseluruhan prioritas. Hal-hal yang dilakukan dalam langkah ini adalah:
   a. Menjumlahkan nilai-nilai dari setiap kolom pada matriks
   b. Membagi setiap nilai dari kolom dengan total kolom yang bersangkutan untuk memperoleh normalisasi matriks. Ini bisa dihitung dengan rumus 2.7.

$$Fij = \frac{Fij}{\sum Fi} \quad \textbf{(2.7)}$$

Dimana : i posisi fitur secara horizontal

j posisi fitur secara vertickal

c. Menjumlahkan nilai-nilai dari setiap baris dan membaginya dengan jumlah elemen untuk mendapatkan nilai rata-rata.

4. Mengukur Konsistensi Dalam pembuatan keputusan, penting untuk mengetahui seberapa baik konsistensi yang ada karena kita tidak menginginkan keputusan berdasarkan pertimbangan dengan konsistensi yang rendah. Hal-hal yang dilakukan dalam langkah ini adalah sebagai berikut:
   a. Kalikan setiap nilai pada kolom pertama dengan prioritas relatif elemen pertama, nilai pada kolom kedua dengan prioritas relatif elemen kedua dan seterusnya.
   b. Jumlahkan setiap baris
   c. Hasil dari penjumlahan baris dibagi dengan elemen prioritas relatif yang bersangkutan.
   d. Jumlahkan hasil bagi di atas dengan banyaknya elemen yang ada, hasilnya disebut λ maks.
5. Hitung Consistency Index (CI) dengan rumus 2.8 :

$$Ci = \frac{\lambda_{maks} - n}{n} \textbf{ (2.8)}$$

dimana n = n merupakan banyaknya elemen

6. Hitung Rasio Konsistensi/Consistency Ratio (CR) dengan rumus 2.9:

$$CR = \frac{CI}{RC} \quad \textbf{(2.9)}$$

Dimana CR = *Consistency Ratio*

CI = Consistency *Index*

IR = *Indeks Random Consistency*

7. Memeriksa konsistensi hierarki. Jika nilainya lebih dari 10%, maka penilaian data *judgment* harus diperbaiki. Namun jika Rasio Konsistensi

(CI/CR) kurang atau sama dengan 0,1, maka hasil perhitungan bisa dinyatakan benar [**15**].

Dimana RI : *random index* yang nilainya dapat dilihat pada table 2.7

**Tabel 2. 7 Ratio Indeks**

| Ukuran Matriks (n) | Indeks Konsistensi Acak (RI) |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 0,52 |
| 4 | 0,89 |
| 5 | 1,11 |
| 6 | 1,25 |
| 7 | 1,35 |
| 8 | 1,40 |
| 9 | 1,45 |
| 10 | 1,49 |

## 2.5 Penggunaan AHP untuk peringkasan text

Seperti yang terlihat pada tabel 2.8, F1,F2,F3,F4,F5,F6 merupakan fitur pada peringkasan text. Pada fitur ini dibuat matriks dimana setiap fitur dibandingkan dengan fitur yang lainnya. Fitur yang dibandingkan dengan fitur yang sama akan menghasilkan nilai 1 [**5**].

**Tabel 2. 8 Perbandingan Fitur**

| | F1 | F2 | F3 | F4 | F5 | F6 | W |
|---|---|---|---|---|---|---|---|
| F1 | 1 | α | | | | | $W_{F1}$ |
| F2 | 1/ α | 1 | | | | | $W_{F2}$ |
| F3 | | | 1 | | | | $W_{F3}$ |
| F4 | | | | 1 | | | $W_{F4}$ |
| F5 | | | | | 1 | | $W_{F5}$ |
| F6 | | | | | | 1 | $W_{F6}$ |

Selanjutnya bandingkan tiap kalimat yang ada pada text dengan perbandingan menggunakan ke 6 fitur. Seperti pada tabel dibawah ini :

**Tabel 2. 9 Perbandingan Kalimat**

| Fke(1..6) | S1 | S2 | ... | Sn | W |
|---|---|---|---|---|---|
| S1 | 1 | α | | | W1 |
| S2 | 1/ α | 1 | | | W2 |
| ... | | | 1 | | ... |
| Sn | | | | 1 | Wn |

Lalu, nilai bobot yang sudah di dapatkan dimasukan kedalam tabel 2.10

**Tabel 2. 10 Tabel bobot setiap Fitur**

| Fke(1..6) | F1 | F2 | F3 | F4 | F5 |
|---|---|---|---|---|---|
| S1 | W11 | W12 | W13 | W14 | W15 |
| S2 | W21 | W22 | W23 | W24 | W25 |
| ... | . | . | . | . | ... |
| Sn | Wn1 | Wn2 | Wn3 | Wn4 | Wn5 |

Bobot (w) untuk setiap fitur (wfj) dan kalimat(wij) yang diperoleh dengan cara perbandingan berpasangan akan dihitung kembali bobotnya dengan cara dikalikan antara fitur (wfj) dengan kalimat(wij) dengan rumus :

$$W_{Si} = \sum_{j=1}^{6} W_{Fj} * W_{ij} \quad \textbf{(2.10)}$$

## 2.6 Teknik Evaluasi Peringkasan Teks Rouge

Rouge singkatan dari Recall-Oriented Understudy for Gisting Evaluation, Rouge merupakan metode untuk melakukan evaluasi peringkasan teks. Metode ini bekerja dengan cara membandingkan ringkasan yang dihasilkan sistem dengan ringkasan yang dihasilkan secara manual. Metode Rouge terbagi atas 3 macam diantaranya adalah Rouge-N, Rouge-L, dan Rouge W. Pada penelitian ini akan menggunakan Rouge-N. Rouge-N merupakan recall n-gram antara ringkasan manual dengan ringkasan pada sistem.

Rumus pada temu kembali informasi adalah

$$R = \frac{\{relevan\} \cap \{ditemukan\}}{\{relevan\}} \quad (2.11)$$

Pada Rouge-N yang bertindak sebagai {relevan} merupakan n-gram pada kalimat ringkasan secara manual, sedangkan yang bertindak sebagai {ditemukan} adalah n-gram pada ringkasan sistem. Dimana n-gram merupakan jumlah range kata yang dipakai. Pada penelitian ini nilai n-gram adalah 1.

Rumus yang dipakai dalam teknik Rouge untuk menghitung nilai Rouge adalah

$$R_{AiBj} = \frac{Clipped\ trigram_{AiBj}}{Count\ trigram_{Ai}} \quad \textbf{(2.12)}$$

$$P_{AiBj} = \frac{Clipped\ trigram_{AiBj}}{Count\ trigram_{Bi}} \textbf{(2.13)}$$

$$F_{AiBj} = \frac{(1+\beta^2)R_{AiBj}}{R_{AiBj} + \beta^2 P_{AiBj}} \textbf{(2.14)}$$

R merupakan kemampuan sistem untuk memanggil dokumen yang relevan. P merupakan kemampuan sistem untuk tidak memanggil dokumen yang tidak relevan. Sedangkan F merupakan evaluasi hasil ringkasan yang dihasilkan R dan P. $Clipped\ trigram_{AiBj}$ merupakan jumlah kata yang sama antara kalimat 1 dengan kalimat 2. Sedangkan $Count\ trigram_{Ai}$merupakan jumlah kata pada kalimat 1 dan $Count\ trigram_{Bi}$ merupakan jumlah kata pada kalimat b [**16**]. Nilai $\beta$ adalah nilai penyeimbang antara nilai Recall dan precision. Nilai $\beta$ pada penelitian ini adalah 1 [**17**].

## 2.7. Perangkat pemodelan Analisis terstruktur

Perangkat pemodelan analisis terstruktur merupakan alat bantu pemodelan yang digunakan untuk menggambarkan hasil pelaksanaan analisis terstruktur. Pada perangkat pemodelan analisis terstrukur ini terbagi atas :

1. Diagram Konteks

Diagram konteks merupakan diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem [**18**]. Diagram konteks ini merupakan proses pertama yang ada di DFD biasanya digambarkan dengan bulatan dan juga menggambarkan proses input dan output dari sistem. Diagram konteks ini akan menggambarkan tentang keseluruhan sistem.

2. *Data Flow Diagram* (DFD)

Diagram ini berfungsi untuk menggambarkan aliran data dalam sistem, baik itu sumber dan tujan data, proses yang mengolah data tersebut dan tempat penyimpanan datanya. Manfaat dari DFD ini adalah untuk memungkingkan pengembangan sistem dari DFD level tinggi ke DFD level rendah. Beberapa simbol yang dapat gunakan didalam DFD antara lain :

a. Entitas Luar

Berupa obyek yang memberikan data kedalam sistem atau memberikan data dari sistem, biasanya disimbolkan dengan suatu kotak. Entitas ini bukan merupakan bagian dari sistem.

b. Proses

Proses merupakan apa yang dikerjakan oleh sistem. Proses atau aliran data masuk menjadi aliran data keluar. Proses ini berfungsi untuk mentransformasikan suatu atau beberapa data keluaran sesuai dengan spesifikasi yang diinginkan, biasanya digambarkan dengan bentuk bulat.

c. Arus data

Arus data ini merupakan tempat mengalirkan informasi dan juga menggambarkan komponen mana saja yang saling terhubung, biasanya digambarkan dengan bentuk arah panah dan diberi nama atas arus data yang mengalir.

d. Simpanan Data

Simpanan data ini merupakan tempate *store* atau penyimpanan data yang ada didalam sistem. Simpanan data ini biasanya digambarkan dengan dua garis sejajar.

3. Kamus Data

Menurut Rosa dan Shalahudin [**19**], kamus data dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. Kamus data juga merupakan daftar elemen data yang mengalir pada sistem perangkat lunak baik itu masukan (*input*), dan keluaran (*output*) bisa dipahami. Kamus data biasanya berisi :

a. Nama : nama dari data
b. Digunakan pada : biasanya proses – proses terkait data
c. Deskripsi : deskripsi data
d. Informasi tambahan seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.

4. *Flow Chart*

Menurut Supardi [**20**] *Flow Chart* merupakan bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.

Menurut Jogiyanto [**21**] terdapat lima macam bagan alir, yaitu sebagai berikut :

1. Bagan Alir Sistem ( *Systems Flowchart* )

Merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan. Bagan ini menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem.

2. Bagan Alir Dokumen *( Document Flowchart )*

Bagan alir dokumen atau disebut juga bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. Bagan alir dokumen ini menggunakan simbol-simbol yang sama dengan yang digunakan di dalam bagan alir sistem.

3. Bagan Alir Skematik ( *Schematic Flowchart* )

Merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur didalam sistem. Perbedaanya adalah, bagan alir skematik selain menggunakan simbol-simbol bagan alir sistem juga menggunakan gambar-gambar komputer dan peralatan lainnya yang digunakan. Maksud penggunaan gambar-gambar ini adalah untuk memudahkan komunikasi kepada orang yang kurang pahan dengan simbol-

simbol bagan alir. Penggunaan gambar-gambar ini memudahkan untuk dipahami, tetapi sulit dan lama menggambarnya.

4. Bagan Alir Program ( *Program Flowchart* )

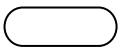
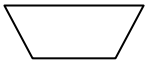
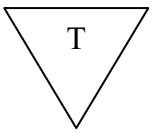
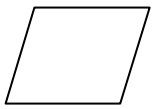
Merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dapat terdiri dari dua macam, yaitu Bagan Alir Logika Program ( *Program Logic Flowchart* ) dan Bagan Alir Program Komputer terinci ( *Detailed Computer Program Flowchart* ).

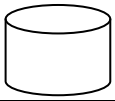
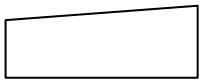
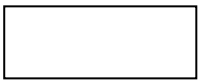
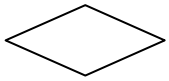
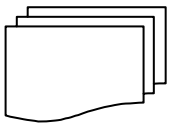
5. Bagan Alir Proses ( *Process Flowchart* )

Merupakan bagan alir yang banyak digunakan di teknik industri. Bagan alir ini juga berguna bagi analis sistem untuk menggambarkan proses dalam suatu prosedur. Bagan alir proses selain dapat menunjukkan kegiatan dan simpanan yang digunakan dalam suatu prosedur, dapat juga menunjukkan jarak kegiatan yang satu dengan yang lainnya serta waktu yang diperlukan oleh suatu kegiatan.

Menurut Krismiaji [**22**] simbol dari bagan alir ( *flowchart* ) bisa dilihat pada tabel 2.11.

**Tabel 2. 11 Simbol *Flowchart***

| No | Simbol | Pengertian | Keterangan |
|---|---|---|---|
| 1. |  | Mulai / berakhir ( *Terminal* ) | Digunakan untuk memulai, mengakhiri, atau titik henti dalam sebuah proses atau program; juga digunakan untuk menunjukkan pihak eksternal. |
| 2. |  | Dokumen | Sebuah dokumen atau laporan; dokumen dapat dibuat dengan tangan atau dicetak oleh komputer. |
| 3. |  | Kegiatan Manual | Sebuah kegiatan pemrosesan yang dilaksanakan secara manual. |
| 4. |   | Arsip | Arsip dokumen disimpan dan diambil secara manual. Huruf didalamnya menunjukkan cara pengurutan arsip: N = Urut Nomor; A = Urut Abjad;T = Urut Tanggal. |
| 5. |  | Input / Output; Jurnal / Buku Besar | Digunakan untuk menggambarkan berbagai media input dan output dalam sebuah bagan alir program. |

| | | | |
|---|---|---|---|
| 6. |  | Disk Bermagnit | Data disimpan secara permanen pada disk bermagnit. |
| 7. |  | Penghubung Pada Halaman Berbeda | Menghubungkan bagan alir yang berada dihalaman yang berbeda. |
| 8. |  | Pemasukan Data On Line | Entri data alat oleh on line seperti terminal CRT dan komputer pribadi. |
| 9. |  | Pemrosesan Komputer | Sebuah fungsi pemrosesan yang dilaksanakan oleh komputer biasanya menghasilkan perubahan terhadap data atau informasi |
| 10. |  | Arus Dokumen atau Pemrosesan | Arus dokumen atau pemrosesan; arus normal adalah ke kanan atau ke bawah. |
| 11. |  | Keputusan | Sebuah tahap pembuatan keputusan |
| 12. |  | Penghubung Dalam Sebuah Halaman | Menghubungkan bagan alir yang berada pada halaman yang sama. |
| 13. |  | Dokumen Rangkap | Digambarkan dengan menupuk simbol dokumen dan pencetakan nomor dokumen dibagian depan dokumen pada bagian kiri atas. |