PERINGKASAN ARTIKEL BERBAHASA INDONESIA MENGGUNAKAN *TEXTRANK* DENGAN PEMBOBOTAN BM25

PROPOSAL SKRIPSI

Disusun oleh:
Yurdha Fadhila Hernawan
NIM: 165150200111094

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2019

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Penggunaan internet sebagai sumber informasi telah membawa manusia pada era *one click away*. Apa pun bisa diakses di mana pun kapan pun, baik secara visual maupun tidak. Namun, apakah setiap informasi yang diakses selalu sesuai dengan konteks yang diinginkan? Bisa dikatakan hanya sedikit pengguna yang dapat memahami semua informasi ketika membaca sebuah tulisan panjang (Niu et al., 2016). Kesulitan tersebut akan membuat pengguna untuk membaca ulang, sehingga akan menghabiskan banyak waktu.

Untuk memudahkan pengguna internet dalam mendapatkan informasi yang ringkas dengan tidak merusak atau menghilangkan informasi penting, maka dibutuhkan suatu peringkasan otomatis (Abbasi-ghalehtaki et al., 2016). Berdasarkan hasil ringkasannya, peringkasan teks dapat dikategorikan dalam dua bentuk, yaitu ekstraktif dan abstraktif. Ringkasan ekstraktif merupakan ringkasan yang terdiri atas kumpulan dari bagian-bagian penting suatu tulisan yang dapat mewakili keseluruhan teks, sedangkan ringkasan abstraktif merupakan ringkasan yang terdiri dari kalimat baru yang dapat merepresentasikan konteks tulisan dalam bentuk lain. Selain itu, peringkasan teks juga dapat dikelompokkan berdasarkan dokumen yang digunakan menjadi *single document* dan *multi-document* (Fang et al., 2017).

Masalah utama yang muncul setelah melakukan peringkasan adalah kualitas hasil peringkasan. Apakah konteks yang dibicarakan pada hasil ringkasan sudah dapat merepresentasikan tulisan secara utuh. Penelitian sebelumnya yang melakukan peringkasan ekstraktif teks berbahasa Indonesia memanfaatkan kata benda yang terdapat dalam sebuah dokumen (Pinandhita, 2013). Penelitian tersebut menghitung nilai kemiripan (*similarity*) antara kalimat berdasarkan kata benda yang terdapat pada setiap kalimat, lalu memeringkatkan kalimat yang paling penting dengan mengurutkan total nilai kemiripan tersebut. Evaluasi dilakukan dengan membandingkan hasil kinerja beberapa metode *similarity* dengan hasil ringkasan yang didapatkan dari penelitian Miptahudin (2010) dan Aristoteles (2011).

Selain itu, (Niu et al., 2016) telah melakukan penelitian untuk mendapatkan ringkasan secara abstraktif. Penelitian tersebut menggunakan dokumen teks opini pendek berbahasa China dengan mengelompokkan teks yang mirip lalu meringkasnya. Proses pengelompokan teks yang mirip dilakukan dengan menggunakan metode *K-Means* dengan fitur yang didapatkan dari nilai *word2vec* (nilai hubungan kedekatan antara kata). Hasil pengelompokan teks akan diperingkatkan menggunakan metode *TextRank.* Kalimat yang dianggap yang dianggap penting dan dapat dijadikan ringkasan memiliki peringkat tertinggi. Hasil pemeringkatan akan dijadikan ringkasan dengan menggunakan *encoder-decoder Reccurent Neural Network (RNN)* untuk membentuk kalimat baru.

Berdasarkan pemaparan di atas, penulis mengajukan penelitian untuk melakukan peringkasan otomatis dengan objek artikel berita *online* berbahasa Indonesia menggunakan metode pemeringkatan *TextRank* dan BM25 sebagai fungsi *similarity.* BM25 dipilih berdasarkan penelitian sebelumnya yang telah melakukan pengujian menggunakan beberapa fungsi *similarity* dan mendapatkan bahwa BM25 menghasilkan ringkasan yang lebih baik dari pada fungsi *similarity* lainnya (Pinandhita, 2013).

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah disebutkan, dapat dirumuskan masalah sebagai berikut:

1. Apakah *TextRank* dapat menghasilkan ringkasan ekstraktif otomatis dengan kualitas yang baik?

1.3 Tujuan

1. Menghasilkan ringkasan ekstraktif secara otomatis dengan kualitas yang baik menggunakan *TextRank*

1.4 Manfaat

Selain menghasilkan ringkasan secara otomatis, penelitian ini dapat digunakan dalam *Question Answering System, Information retrival*, dan ekstraksi informasi.

1.5 Batasan Masalah

1. Artikel online yang digunakan adalah artikel berita berbahasa Indonesia yang memiliki tulisan sesuai Ejaan Bahasa Indonesia (EBI)
2. Artikel online berasal dari situs resmi BBC Indonesia dengan topik bencana alam sebanyak 10 buah.

1.6 Sistematika Pembahasan

Laporan penelitian ini terdiri atas beberapa bagian yaitu sebagai berikut.

Bab I Pendahuluan

Berisikan latar belakang penelitian, rumusan masalah, tujuan penelitian, manfaat, batasan masalah yang diberikan, dan sistematika pembahasan.

Bab II Landasan Kepustakaan

Berisikan pembahasan mengenai teori yang dijadikan pedoman pengerjaan, konsep pengerjaan penelitian, metode atau sistem yang diterapkan, pustaka ilmiah yang berkitan dengan *Natural Language Processing, Text Summarize, dan TextRank*.

Bab III Metode Penelitian

Berisikan langkah pengerjaan dalam penelitian, teknik yang digunakan, data yang akan digunakan, dan representasi berdasarkan metode yang dipilih untu menyelesaikan masalah.

Bab IV Perancangan

Berisikan proses penyelesaian masalah dengan visualisasi diagram alir dan perhitungan manual dari metode yang digunakan.

Bab V Implementasi

Berisikan implementasi metode yang digunakan dalam menyelesaikan masalah dalam sebuah sistem.

Bab VI Pengujian dan Analisis

Berisikan scenario pengujian dan analisis terhadap hasil pengujian.

Bab VII Penutup

Berisikan kesimpulan dan saran sebagai rekomendasi untuk penelitian selanjutnya.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Salah satu penelitian yang dilakukan untuk melakukan peringkasan adalah dengan memanfaatkan kata benda yang terdapat pada sebuah dokumen (Pinandhita, 2013). Penelitian tersebut dilakukan untuk mendapatkan ringkasan ekstraktif dengan membandingkan beberapa metode *similarity* yang didapatkan dari kata benda. Dokumen yang digunakan berupa artikel koran dengan topik di luar pertanian yang juga digunakan pada penelitian Miptahudin (2010) dan dokumen dari penelitian Aristoteles (2011). Hasil peringkasan yang didapatkan dari beberapa metode *similarity* yang digunakan Pinandhita akan dibandingkan dengan hasil peringkasan yang didapatkan Miptahudin (2010) dan Aristoteles (2011).

Total percobaan yang dilakukan Pinandhita adalah sebanyak tujuh kali (dengan mengikutsertakan judul dokumen dan tanpa judul dokumen) yaitu penerapan *PageRank* dengan bobot *cosine similarity*, *PageRank* dengan bobot BM25, *PageRank* dengan bobot *content overlap*, bobot *cosine similarity* tanpa PageRank, bobot BM25 tanpa *PageRank*, bobot *content overlap* tanpa *PageRank*, dan yang terakhir menggunakan bobot koefisien *dice*. Rata rata panjang dokumen yang digunakan pada penelitian ini adalah sebanyak 47 kalimat yang mengandung rata rata sebanyak 282 kata benda. Pengujian dilakukan dengan mencari nilai *kappa* berdasarkan kesepakatan hasil ringkasan antara tiga orang dosen Jurusan Sastra Indonesia dan tiga orang mahasiswa. Kualitas ringkasan terbaik didapatkan dari hasil pembobotan BM25 dengan *PageRank* tanpa menggunakan judul dokumen.

Selain mendapatkan ringkasan secara ekstraktif, peneliti lain juga telah melakukan penelitian untuk mendapatkan ringkasan secara abstraktif (Niu et al., 2016). Penelitian tersebut menggunakan dokumen teks opini pendek berbahasa China dengan mengelompokkan teks yang mirip lalu meringkasnya. Proses peringkasan dibagi atas tiga tahap yaitu *clustering* teks (mengelompokkan), pemeringkatan teks, dan peringkasan teks. Proses *clustering* dilakukan dengan menggunakan metode *K-Means* dengan fitur yang telah didapatkan dari nilai *word2vec*. *Word2vec* menunjukan hubungan kedekatan antara suatu kata dengan kata lainnya.

Hasil *clustering* yang telah didapatkan akan diperingkatkan menggunakan model pemeringkatan graf yaitu *textrank* dengan bantuan pembobotan BM25, setelahnya hasil pemeringkatan tertinggi akan diringkas secara abstraktif. Peringkasan abstraktif dilakukan dengan metode *encoder-decoder Reccurent Neural Network (RNN)*. Pengujian kualitas ringkasan yang dihasilkan pada penelitian ini dilakukan dengan mencari nilai *precision, recall, F-measure, ROUGE-N, dan ROUGE-L*. Secara berurut rata-rata nilai *precision, recall, dan F-measure* yang didapatkan adalah sebesar 0,94; 0,932; dan 0,933. Nilai ROUGE-N didapatkan dari hasil kesamaan *N-gram* yang serupa dari hasil peringkasan manual dengan

hasil ringkasan yang didapatkan sistem. Pada evaluasi ini digunakan ROUGE-1 (*unigrams*) dan ROUGE-2 (*bigrams*).

Peringkasan dokumen bisa dilakukan dengan metode klasifikasi kalimat (Fhadli, 2017). Kalimat akan diklasifikasikan sebagai kelas yang termasuk ringkasan dan kelas yang termasuk bukan ringkasan. Fitur yang digunakan dalam pengklasifikasian ini adalah fitur statistik dan fitur linguistik, sedangkan dokumen yang digunakan dalam penelitian ini adalah literatur Ilmu Komputer Berbahasa Indonesia.

Fitur statistik didapatkan dari nilai TF-IDF sebuah kalimat, sedangkan fitur linguistik didapatkan dari banyaknya kata pada judul yang terdapat pada kalimat (title word), posisi kalimat pada dokumen (sentence location), panjang kalimat (sentence length), kalimat yang mengandung akronim atau singkatan (upper-case word), kalimat yang memiliki frasa yang dianggap penting seperti "jadi", "hasilnya", dan "kesimpulannya" (cue phrase), kalimat yang mengandung kata spesifik yang menyatakan sebuah topik dokumen (biased word), dan kalimat yang mengandung kata-kata yang dianggap tidak penting (occurrence of non-essential information). Proses pengklasifikasian kalimat dilakukan dengan menggunakan metode Naive Bayes. Naive Bayes bekerja dengan cara menghitung peluang suatu kalimat terhadap kelas dengan bantuan data latih. Hasil kualitas ringkasan diuji dengan melakukan pencarian nilai precision, recall, F-measure, dan relative utility. Hasil rata-rata F-measure dan relative utility yang didapatkan adalah 0,206538 dan 0,116657.

Salah satu penelitian yang menjadi acuan adalah pencarian ringkasan secara ekstraktif oleh (Mussina, Aubakirov and Trigo, 2018). Ringkasan yang dihasilkan tidak melakukan perubahan terhadap struktur kalimat. Salah satu cara untuk mendapatkan ringkasan ekstraktif adalah menggunakan data *statistical,* yaitu dengan menghitung kesamaan *text units*. *Text units* tersebut dapat berupa kata, kalimat atau paragraf. Penelitian ini menggunakan dokumen mengenai bencana alam berhasa Rusia dan Kazakh, yang mana memiliki struktur kalimat yang jelas.

Penelitian tersebut merepresentasikan dokumen menggunakan metode *TextRank*¸ dengan kalimat sebagai node dan nilai *similarity* antara kalimat sebagai *edges.* Panjang ringkasan yang akan diambil adalah sebanyak 30% dari total panjang dokumen. Nilai *similarity* yang digunakan adalah *content overlap,* BM25*,* dan *substring* terpanjang yang muncul antara dua kallimat. Namun, penelitian ini tidak menggunakan metode *PageRank* dalam proses pemeringkatan kalimat, melainkan dengan menjumlahkan seluruh nilai *similarity* yang dimiliki kalimat dan mengurutkannya berdasarkan jumlah nilai terbesar. Secara umum, setiap kalimat akan saling berkaitan antara satu dengan yang lainnya, namun terkadang ada kalimat yang tidak memiliki kesamaan dengan kalimat lain. Kalimat tersebut tidak memiliki nilai *similarity* sehingga tidak akan dijadikan ringkasan.

Sebelum melakukan pengambilan ringkasan berdasarkan nilai fungsi *silimarity*, kalimat akan direduksi berdasarkan nilai treshold yang ditetapkan. Kalimat yang

miliki jumlah nilai *similarity* kurang dari nilai treshold tidak akan dijadikan ringkasan. Pengambilan ringkasan akan diambil sebanyak 30% panjang dokumen berdasarkan kalimat yang memiliki nilai *similarity* terbesar. Kalimat terpilih akan dijadikan ringkasan berdasarkan urutan sesuai dokumen awal. Evaluasi pada penelitian tersebut dilakukan dengan mencari nilai distribusi *key-words,* dengan nilai rata-rata untuk nilai *similarity* menggunakan *content overlap, LongestCommonSubstring*, dan BM25 secara berurutan adlah 0,180; 0,175; dan 0,169.

Penelitian lain yang menjadi acuan adalah percobaan untuk membandingkan hasil ringkasan ekstraktif menggunakan beberapa nilai *similarity* dalam *TextRank* (Barrios et al., 2016). Cara mengidentifikasi hubungan antara kalimat satu dengan kalimat lainnya adalah dengan melakukan perhitungan kata yang sama, *cosine distance*, dan kesamaan *query* yang dianggap penting. Pada penelitian tersebut dokumen akan direpresentasikan sebagai graf dengan kalimat sebagai *node* dan nilai *similarity* sebagai *edges* (hubungan antara *nodes*). Fungsi similarity yang digunakan adalah *Longest Common Substring, cosine distance,* BM25*,* dan BM25*+* dengan proses pemeringkatan *PageRank*.

Dokumen yang digunakan adalah *Document Understanding Conference (DUC)* yang berjumlah sebanyak 567 dokumen dengan peringkasan sebanyak 20% dari tiap panjang dokumen. Hasil ringkasan dari percobaan tersebut dievaluasi menggunakan metode *ROUGE-N* dengan nilai terbaik didapatkan pada peringkasan menggunakan BM25 dan BM25*+.*

2.2 Dasar Teori Peringkasan Teks

Ketika jumlah informasi *online* semakin banyak, maka kebutuhan akan sistem yang dapat merangkum satu atau lebih dokumen secara otomatis sangat diperlukan (Radev et al., 2002). Selain dapat membantu dalam mengatasi informasi yang berlebihan, peringkasan otomatis juga berguna dalam penyajian informasi singkat mengingat ukuran perangkat *handy* yang digunakan pembaca (Sankarasubramaniam et al., 2014). Ukuran ringkasan biasanya tidak lebih dari setengah dokumen aslinya. Berdasarkan kebutuhannya ringkasan dokumen terbagi atas (Munot and S. Govilkar, 2013):

1. Metode
    a. Abstraktif

        Ringkasan yang terdiri dari kalimat baru yang dapat merepresentasikan konteks tulisan dalam bentuk kalimat lain.

    b. Ekstraktif

        Ringkasan yang terdiri atas kumpulan dari bagian-bagian penting suatu tulisan yang dapat mewakili keseluruhan teks.
2. Konten
    a. *Generic*

Ringkasan umum yang tidak bergantung pada syarat apapun.

b. *Query based*

Ringkasan hanya didapatkan berdasarkan *query* yang diinginkan pengguna.

3. Jumlah dokumen

a. *Single document*

Ringkasan yang didapatkan dari satu dokumen.

b. *Multi document*

Ringkasan yang didapatkan dari beberapa dokumen.

4. Bahasa

a. *Monolingual*

Ringkasan yang didapatkan dari dokumen dengan bahasa yang sama.

b. *Multilingual*

Ringkasan yang didapatkan dari beberapa dokumen dengan bahasa yang berbeda.

Dalam penelitian ini peringkasan akan dilakukan secara ekstraktif dengan memilih kalimat yang dianggap penting dan dapat dijadikan ringkasan. Dokumen yang digunakan merupakan *single document* dengan berbahasa Indonesia.

2.3 *Preprocessing*

Preprocessing dilakukan sebelum proses pembentukan ringkasan dan mempermudah proses peringkasan. Dalam *preprocessing* terdapat tiga tahapan yaitu segmentasi, *cleaning*, tokenisasi, dan *stemming*.

2.3.1 Segmentasi

Segmentasi merupakan proses pemecahan teks dokumen menjadi kalimat-kalimat untuk mempermudah pemrosesan dokumen menjadi potongan-potongan yang lebih kecil. Pada penelitian yang akan dilakukan, judul pada artikel berita diikutsertakan dalam percobaan guna membantu performa sistem dalam mendapatkan konteks tulisan karna judul dianggap memiliki informasi penting mengenai konteks dari isi dokumen. Contoh hasil segmentasi dokumen ditunjukan pada Tabel 2.1.

Tabel 2.1 Hasil Segmentasi Dokumen

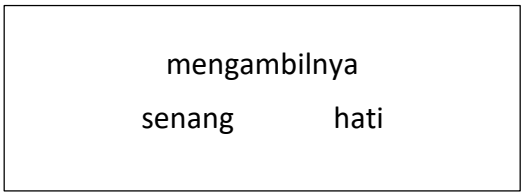
Index	Kalimat
1	Dia mengambilnya dengan senang hati.
2	Esoknya setelah pulang sekolah, ia melarikan diri.
3	Ibu yang bekerja sebagai pembantu.

2.3.2 *Cleaning*

Pada proses cleaning dilakukan penghilangan tanda baca, angka, dan *url* karna dianggap tidak mempengaruhi i

2.3.3 Tokenisasi

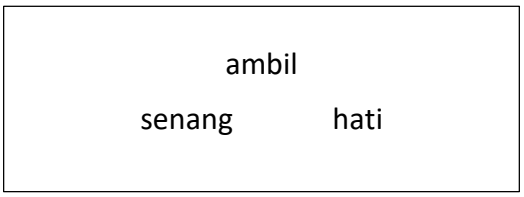
Tokenisasi adalah metode pemecahan teks kalimat menjadi token-token (*term*) yang berurutan. Pada proses ini juga dilakukan penghilangan kata hubung atau kata yang dianggap tidak mempenggaruhi isi dari konten dokumen menggunakan metode *stopwords removal*. Hal ini juga dilakukan guna meningkatkan performa sistem agar sistem bisa secara efektif dimanfaatkan untuk pengolahan konten yang benar-benar dianggap penting saja. Contoh hasil tokenisasi dokumen ditunjukan pada Gambar 2.1.



Gambar 2.1 Hasil Tokenisasi Kalimat Index 1

2.3.4 Stemming

Stemming merupakan metode pembentukan kata atau term menjadi kata dasar. Biasanya proses *stemming* dilakukan dengan membuang imbuhan yang terdapat pada term. Contoh hasil tokenisasi dokumen ditunjukan pada Gambar 2.2.



Gambar 2.2 Hasil *Stemming* Kalimat Index 1

2.4 *Term Weighting*

2.4.1 *Term Frequency (tf)*

Perhitungan *term frequency* merupakan langkah awal dari *term weighting*. *Term frequency* adalah jumlah kemunculan setiap *term* dalam satu kalimat. Contoh hasil perhitungan term frequency ditunjukan padaTabel 2.2.

Tabel 2.2 Hasil Perhitungan *Tf*

Term	**Kalimat 1**	**Kalimat 2**	**Kalimat 3**
ambil	1	0	0
bantu	0	0	1
esok	0	1	0
hati	1	0	0
kerja	0	0	1
lari	0	1	0
pulang	0	1	0
sekolah	0	1	0
senang	1	0	0

2.4.2 *Document Frequency (df)*

Document frequency merupakan jumlah kemunculan setiap *term* dalam satu dokumen. Nilai *document frequency* akan digunakan untuk mendapatkan nilai *Inverse document frequency*.

2.4.3 *Inverse Document Frequency (Idf)*

Nilai *inverse document frequency* akan digunakan untuk perhitungan BM25. Mengacu pada penelitian yang dilakukan (Lv and Zhai, 2011) perhitungan Idf ditunjukan pada Persamaan 2.1 :

$$Idf_t = \log_{10} \frac{N+1}{dft} \qquad (2.1)$$

Keterangan:

N = Panjang kalimat
dft = *Document frequency* tiap *term* dalam satu dokumen

Contoh perhitungan *df* dan *idf* ditunjukan pada

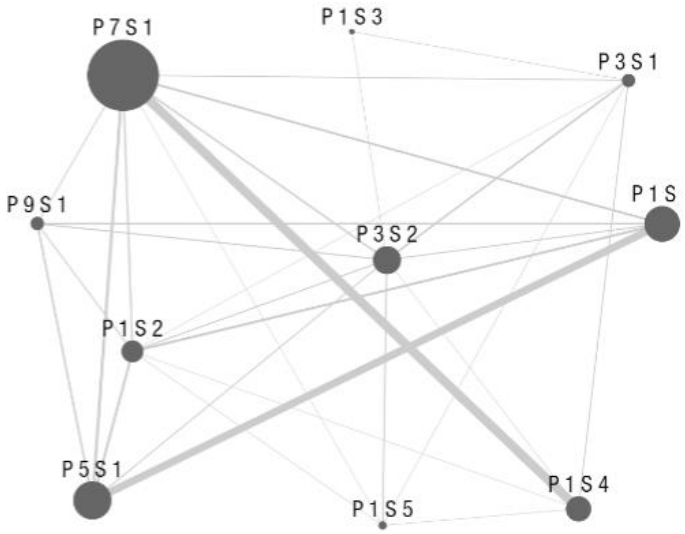
Tabel 2.3 Hasil Perhitungan *Df* dan *Idf*

Term	***df***	***idf***
ambil	1	0,60206
bantu	1	0,60206
esok	1	0,60206
hati	1	0,60206

kerja	1	0,60206
lari	1	0,60206
pulang	1	0,60206
sekolah	1	0,60206
senang	1	0,60206

2.5 *TextRank*

TextRank merupakan sebuah algoritma berbasis graf yang digunakan untuk menentukan *node* mana yang paling penting dalam suatu graf (Tarau, 1973). Struktur *TextRank* ditunjukan pada. Penggunaan *TextRank* dapat dilakukan dalam melakukan penarikan keputusan. Proses melakukan *TextRank* dalam peringkasan teks ekstraktif dimulai dengan mengidentifikasi *single document* yang digunakan. Lalu setiap kalimat direpresentasikan sebagai *node* dan hubungan antara kalimat merupakan fungsi *similarity* yang direpresentasikan sebagai *edges*. *Edges* pada graf dapat memiliki arah maupun tidak. Setelah graf terbentuk, maka selanjutnya akan dilakukan pemeringkatan graf dan mengurutkan *node* yang memiliki nilai pemeringkatan paling tinggi (paling penting). Setelah diurutkan, maka peringkasan dapat diambil berdasarkan peringkat dari kalimat-kalimat tersebut.



Gambar 2.3 Struktur *Textrank* Sebagai Graf

2.6 Fungsi *Similarity* BM25

Dalam merepresentasikan dokumen sebagai graf, *edges* didapatkan dari hasil fungsi *similarity* antar kalimat. Fungsi *similarity* didapatkan dari kemiripan isi kalimat satu dengan kalimat lainnya. Kalimat yang merepresentasikan suatu konteks dalam text akan merekomendasikan kalimat lain yang memiliki konteks yang sama (Tarau, 1973). Fungsi similarity yang digunakan adalah BM25. Nilai BM25 didapatkan dari perhitungan bobot *tf* dan *idf* pada setiap kata (term). Selain

itu juga ditambahkan parameter bebas k_1 dan b dengan nilai k_1 sebesar 1.2 dan b sebesar 0.75 (Manning et al., 2009). Persamaan BM25 dijabarkan pada Persamaan 2.2:

$$RSV_d = \sum_{t \in q} Idf \cdot \frac{(k_1 + 1)\mathrm{tf}_{td}}{k_1\left((1-b) + b\left(\frac{L_d}{L_{ave}}\right)\right) + \mathrm{tf}_{td}} \quad (2.2)$$

Keterangan:

$\sum_{t \in q} Idf$ = Nilai *idf term t*
k_1 dan b = Parameter penskalaan terhadap *tf* dan panjang dokumen
tf_{td} = Frekuensi term t pada kalimat d
L_d dan L_{ave} = Panjang kalimat *d* dan rata-rata dari panjang seluruh koleksi kalimat

2.7 *PageRank*

PageRank adalah metode yang digunakan dalam pemeringkatan graf. *PageRank* digunakan oleh Google untuk menentukan tingkat kepentingan halaman web. *PageRank* merupakan nilai mumerical yang menyatakan seberapa penting sebuah halaman web di internet. Singkatnya, perhitungan nilai tersebut bertambah bila halaman tersebut muncul sebagai sebuah *hyperlink* di sebuah halaman web lainnya. Semakin besar nilai yang dimiliki, maka semakin penting web tersebut. Begitu juga dengan kalimat yang saling berhubungan satu sama lain dalam sebuah graf. Kalimat yang penting akan memiliki nilai *PageRank* yang besar.

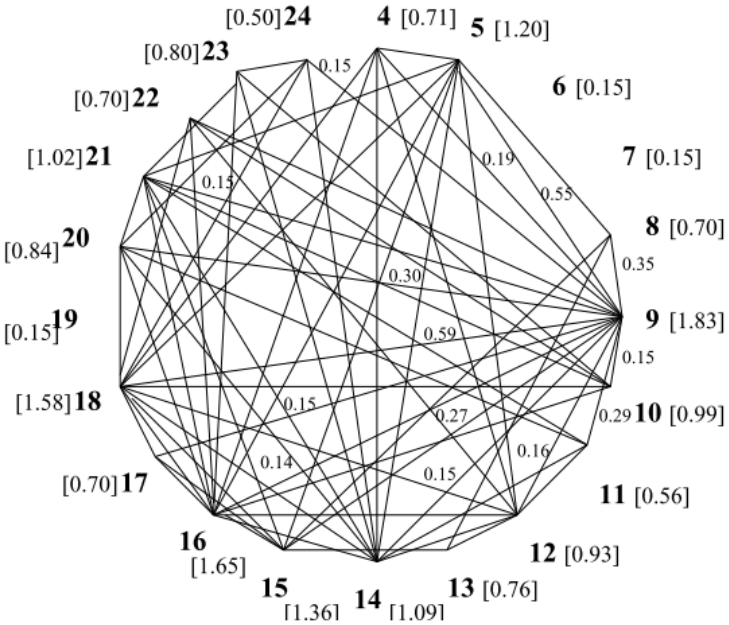
Inisialisasi awal nilai *PageRank* tiap kalimat ditentukan secara random mulai dari 0 hingga 1. Lalu sejumlah iterasi dilakukan untuk melakukan *update* bobot *PageRank* ditiap kalimat. Persamaan *PageRank* dijabarkan pada Persamaan 2.3:

$$WS(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(Vj)} w_{jk}} WS(V_j) \quad (2.3)$$

Keterangan:

$WS(V_i)$ = Bobot nilai *PageRank* kalimat i
d = *Dampening factor*
$V_j \in In(V_i)$ = Kalimat *j* yang berhubungan dengan kalimat *i*
w_{ji} = Nilai fungsi *similarity* antara kalimat *j* dan *i*
$V_k \in Out(Vj)$ = Kalimat *k* yang berhubungan dengan kalimat *j*
w_{jk} = Nilai fungsi *similarity* antara kalimat *j* dan *k*
$WS(V_j)$ = Bobot nilai *PageRank* kalimat j

Dampening factor (*d*) adalah nilai yang telah dihitung oleh Google Engineers dalam sistem PageRank untuk memastikan bahwa bobot *node* akan konvergen pada satu nilai. Nilai *dampening factor* bisa didapatkan dari angka random mulai dari nol hingga satu, namun 0.85 telah menjadi nilai yang umum saat menetapkan nilai *dampening factor.* Pada akhir perhitungan graf kalimat dapat diilustrasikan seperti pada Gambar 2.4 .



Gambar 2.4 Struktur *Textrank* Setelah Perhitungan *Pagerank*

2.8 Evaluasi

Pengujian sistem peringkasan ini akan dilakukan dengan membandingkan nilai *precision, recall,* dan *f-measure* untuk setiap hasil ringkasan. Proses perhitungan tersebut dibantu dengan menggunakan *confussion matrix*.

a. *Confussion Matrix*

Confusion Matrix merupakan informasi mengenai klasifikasi aktual dan prediksi yang dilakukan oleh sistem klasifikasi. Contoh confusion matrix ditunjukkan pada Tabel 2.4.

Tabel 2.4 Contoh *Confusion Matrix*

	True label A	***True not A***
Predicited label A	*True positive (tp)*	*False positive (fp)*
Predicited not A	*False negative (fn)*	*True negative (tn)*

Keterangan :

True Positive = Semua data positif yang dianggap positif oleh sistem

False Positive = Semua data negatif yang dianggap positif oleh sistem

False Negative = Semua data positif yang dianggap negatif oleh sistem

True Negative = Semua data negative yang dianggap negaif oleh sistem

b. Precision

Merupakan nilai ketepatan antara informasi yang dihasilkan sistem dengan hasil informasi yang seharusnya (dianggap benar). Persamaan *precision* ditunjukan pada Persamaan 2.5:

$$Precision = \frac{TP}{TP + FP} \tag{2.5}$$

Keterangan:
TP = True Positive
FP = False Positive

c. Recall

Merupakan tingkat keberhasilan sistem dalam peringkasan otomatis yang menentukan berapa proporsi kalimat yang dipilih oleh pakar yang juga dipilih oleh sistem. Persamaan *recall* ditunjukan pada Persamaan 2.6:

$$Recall = \frac{TP}{TP + FN} \tag{2.6}$$

Keterangan:
TP = True Positive
FN = False Negative

d. F-Measure

Merupakan pengukuran yang menilai timbal balik antara *precision* dan *recall.* Persamaan *F-Measure* ditunjukan pada Persamaan 2.7:

$$F - measure = \frac{2\ x\ P\ x\ R}{P + R} \tag{2.7}$$

Keterangan:
P = Precision
R = Recall

BAB 3 METODOLOGI

3.1 Jenis Penelitian

Penelitian yang diajukan bersifat Non-implementatif dengan jenis penelitian eksperimen. Yang mana hasil dari penelitian ini akan dinilai oleh pakar untuk mengetahui kualitas ringkasan yang dihasilkan dari fungsi *similarity* yang digunakan. Hasil tersebut akan dibandingkan dan dianalisis menggunakan metode evaluasi.

3.2 Pengambilan Data

Jumlah populasi pada penelitian ini termasuk ke dalam kategori tak terbatas karna dokumen yang digunakan merupakan artikel berita online berbahasa Indonesia. Populasi tak terbatas yaitu sumber datanya tidak dapat ditentukan batasan-batasannya sehingga relatif tidak dapat ditentukan dalam bentuk angka. Teknik pengambilan sampel yang digunakan adalah *simple random sampling*. Yaitu pengambilan sampel dari anggota populasi secara acak tanpa memperhatikan tingkatan, karena populasi bersifat homogen (sama).

Instrumen pengumpulan dokumen pada penelitian ini adalah peneliti sendiri, yang mana dokumen diambil dari artikel berita online BBC Indonesia yang dapat diakses melalui situs resmi BBC Indonesia. Jumlah dokumen yang akan diambil sebagai dokumen pengujian sistem adalah sebanyak 10 artikel berita mengenai bencana alam.

3.3 Metode Penelitian

Penelitian ini terbagi atas dua tahap, yaitu pembentukan ringkasan dan evaluasi ringkasan yang dihasilkan. Tahap pembentukan ringkasan terdiri atas tahapan *preprocessing*, *term weighting,* penerapan *TextRank dan PageRank,* dan pembentukan ringkasan. Hasil ringkasan yang didapatkan dari penelitian ini adalah kalimat yang memiliki nilai *PageRank* yang besar setelah diurutkan. Kalimat tersebut dianggap penting dan dapat mereprenstasikan keseluruhan dokumen sehingga dapat dijadikan ringkasan. Mengacu pada penelitian yang dilakukan (Yeh et al., 2005) total kalimat yang akan diambil sebagai ringkasan ialah sebanyak 5%-30% dari total keseluruhan kalimat. Secara umum, diagram proses peringkasan teks pada penelitian ini ditunjukan pada Gambar 3.1.

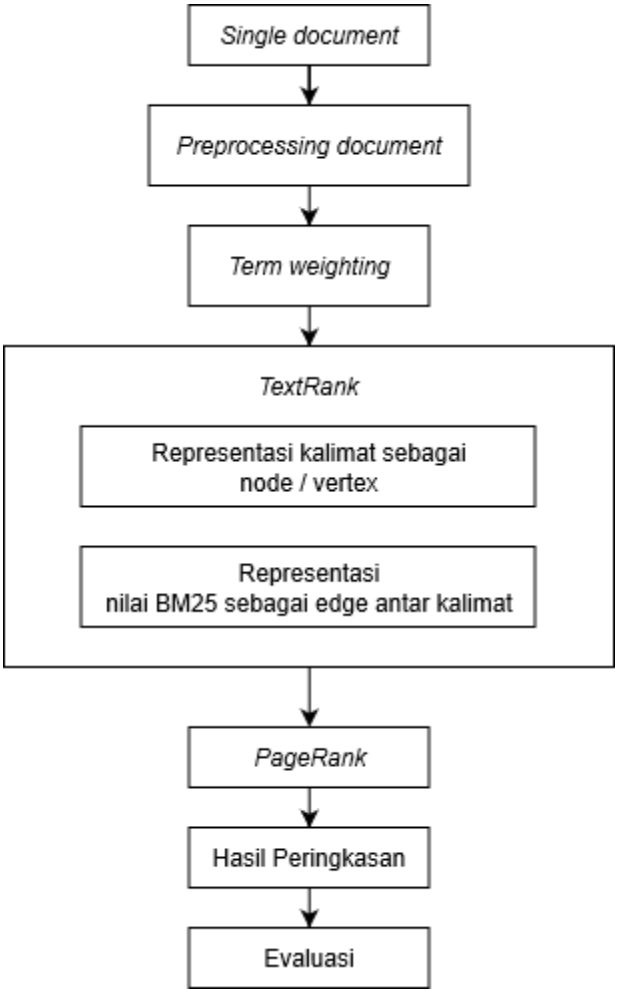
3.4 Peralatan Pendukung

Lingkungan implementasi system *hardware* meliputi:

1. Processor 2.00 GHz
2. RAM 8.00 GB

Spesifikasi *software* meliputi:

1. Operating System Windows 10 64-bit
2. Jupyter Notebook



Gambar 3.1 Gambaran Umum Proses Peringkasan *Single Document*

3.5 Teknik Analisis Data

Pengujian hasil ringkasan yang dihasilkan sistem akan dilakukan oleh dosen Bahasa Indonesia dari Fakultas Ilmu Budaya Universitas Brawijaya. Dosen Bahasa Indonesia tersebut dijadikan pakar dan diminta untuk memilih kalimat yang dianggap penting dan dapat dijadikan ringkasan pada sekumpulan dokumen yang diujikan. Hasil pemilihan kalimat pada tiap dokumen akan dibandingkan dengan hasil peringkasan otomatis yang didapatkan oleh sistem yang dibuat menggunakan *textrank.* Perhitungan *precision, recall,* dan *F-Measure* digunakan untuk mengetahui kualitas ringkasan otomatis tersebut.

DAFTAR RUJUKAN

Abbasi-ghalehtaki, R., Khotanlou, H. and Esmaeilpour, M., 2016. Fuzzy evolutionary cellular learning automata model for text summarization. *Swarm and Evolutionary Computation*, [online] 30, pp.11–26. Available at: <http://dx.doi.org/10.1016/j.swevo.2016.03.004>.

Barrios, F., López, F., Argerich, L. and Wachenchauzer, R., 2016. Variations of the Similarity Function of TextRank for Automated Summarization. [online] Available at: <http://arxiv.org/abs/1602.03606>.

Fang, C., Mu, D., Deng, Z. and Wu, Z., 2017. Word-sentence co-ranking for automatic extractive text summarization. *Expert Systems with Applications*, [online] 72, pp.189–195. Available at: <http://dx.doi.org/10.1016/j.eswa.2016.12.021>.

Fhadli, M., Fauzi, M.A. and Afirianto, T., 2017. Peringkasan Literatur Ilmu Komputer Bahasa Indonesia Berbasis Fitur Statistik dan Linguistik menggunakan Metode Gaussian Naïve Bayes. 1(4), pp.307–319.

Lv, Y. and Zhai, C., 2011. Lower-bounding term frequency normalization. *International Conference on Information and Knowledge Management, Proceedings*, pp.7–16.

Munot, N. and S. Govilkar, S., 2013. Comparative Study of Text Summarization in Indian Languages. *International Journal of Computer Applications*, 75(6), pp.17–21.

Mussina, A., Aubakirov, S. and Trigo, P., 2018. Automatic Document Summarization based on Statistical Information. (Data), pp.71–76.

Niu, J., Zhao, Q., Wang, L., Chen, H., Atiquzzaman, M. and Peng, F., 2016. OnSeS: A novel online short text summarization based on BM25 and neural network. *2016 IEEE Global Communications Conference, GLOBECOM 2016 - Proceedings*, pp.1–6.

Pinandhita, R.R., 2013. Peringkas Dokumen Berbahasa Indonesia Berbasis Kata Benda Dengan BM25.

Radev, D. R., Hovy, E., & McKeown, K., 2002. Introduction to the special issue on summarization. *Computational Linguistics*, 28(4), pp.399–408.

Sankarasubramaniam, Y., Ramanathan, K. and Ghosh, S., 2014. Text summarization using Wikipedia. *Information Processing and Management*, [online] 50(3), pp.443–461. Available at: <http://dx.doi.org/10.1016/j.ipm.2014.02.001>.

Tarau, R.M. and P., 1973. TextRank: Bringing Order into Texts. *Comparative Biochemistry and Physiology -- Part B: Biochemistry and*, [online] 45(4). Available at: <http://www.aclweb.org/anthology/W04-3252>.

Yeh, J.Y., Ke, H.R., Yang, W.P. and Meng, I.H., 2005. Text summarization using a

trainable summarizer and latent semantic analysis. *Information Processing and Management*, 41(1), pp.75–95.