

Peringkasan Teks Otomatis Pada Artikel Berita Hiburan Berbahasa Indonesia Menggunakan Metode BM25

Desy Andriani¹, Indriati², Muhammad Tanzil Furqon³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹andrianidsy@gmail.com, ²indriati.tif@ub.ac.id, ³m.tanzil.furqon@ub.ac.id

Abstrak

Salah satu kegiatan yang paling sering dilakukan pengguna internet Indonesia adalah membaca berita. Lebih dari 50% pengguna internet Indonesia memanfaatkan internet untuk membaca berita. Namun, masalah akan muncul apabila isi dari artikel adalah teks yang panjang sehingga pembacanya membutuhkan waktu untuk membaca dan memahami isi dari artikel tersebut. Salah satu cara agar pengguna tetap dapat membaca dan memahami isi artikel dengan cepat adalah dengan membaca ringkasannya. Maka dari itu diperlukan sistem peringkasan teks otomatis secara ekstratif pada artikel berita hiburan dengan tujuan menekankan informasi utama dan membantu pembaca memperoleh informasi utama dari teks dengan cepat tanpa perlu membaca seluruh isi teks atau dokumen. Penelitian ini menggunakan metode BM25 yang merupakan metode pembobotan kalimat yang mengurutkan kalimat berdasarkan *term* yang muncul pada setiap kalimat dalam dokumen. BM25 menggunakan pembobotan *tf idf* untuk pembobotan kata dan hubungan antara *term* dan tiap kalimat dalam dokumen dipengaruhi oleh parameter bebas k_1 dan b . Berdasarkan hasil pengujian didapatkan bahwa peringkasan teks dengan metode BM25 memperoleh nilai rata-rata *precision*, *recall*, dan *f-measure* terbaik saat nilai *compression rate* yang digunakan adalah 30%. Dimana nilai rata-rata *precision*, *recall*, dan *f-measure* secara berurutan adalah sebesar 0,730, 0,738, dan 0,734.

Kata kunci: peringkasan teks otomatis, text mining, BM25, compression rate

Abstract

One of the most often activitiy carried out by Indonesian internet users is reading news. More than 50% of Indonesian internet users use the internet to read news. However, problems will arise if the content of the article is a long text so that the reader needs time to read and understand the contents of the article. One way that users can still read and understand the contents of articles quickly is by reading the summary. Therefore we need an automatic text summarization system in entertainment news articles with the aim of emphasizing the main information and helping the reader get the main information from the text quickly and don't need to read the entire contents of the text or document. This study uses the BM25 method which is a method of weighting sentences that sort sentences based on terms that appear in each sentence in the document. BM25 is using tf idf weighting for word weighting and the relationship between terms and each sentence in the document is influenced by free parameters k_1 and b . Based on the test results it was found that summarizing the text with the BM25 method obtained the best average precision result, recall and f-measure values when the value of the compression rate used was 30%. Where the average values of precision, recall, and f-measure are 0,730, 0,738 and 0,734.

Keywords: automatic text summarization, text mining, BM25, compression rate

1. PENDAHULUAN

Kebutuhan manusia akan internet terus meningkat. Dalam empat dekade, internet terus berkembang dari hanya digunakan oleh kumpulan peneliti ilmu komputer menjadi tumpuan bagi informasi masyarakat global, dan

saat ini lebih dari satu miliar orang menggunakan internet baik untuk berkomunikasi, mencari dan berbagi informasi, melakukan bisnis, dan menikmati hiburan.

Menurut survey yang dilakukan oleh Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), pengguna internet di Indonesia pada

tahun 2017 adalah sebanyak 143,26 juta dari total populasi 262 juta orang. Hal ini menunjukkan lebih dari setengah populasi masyarakat Indonesia adalah pengguna internet (APJII, 2017).

Selain itu, hasil penelitian APJII juga menunjukkan tentang perilaku pengguna internet Indonesia. Pada bidang gaya hidup, terdapat tujuh kegiatan yang paling sering dilakukan oleh pengguna. Salah satunya adalah membaca berita hiburan atau *entertainment*. Dimana kegiatan membaca berita ini mempunyai persentase yang besar yaitu sebanyak 58,01%. Dari hasil penelitian APJII, diketahui bahwa manusia tidak bisa lepas melakukan kegiatan membaca. Lebih dari 50% pengguna internet Indonesia memanfaatkan internet untuk membaca artikel berita. Namun, masalah akan muncul apabila isi dari artikel adalah teks yang panjang sehingga pembacanya membutuhkan waktu untuk membaca dan memahami isi dari artikel tersebut. Padahal para pengguna tidak selalu memiliki banyak waktu dikarenakan terkadang pengguna mempunyai kesibukan sehingga tidak dapat membaca teks yang cukup panjang. Salah satu cara agar pengguna tetap dapat membaca dan memahami isi artikel dengan cepat adalah dengan membaca ringkasannya. Maka dari itu diperlukan sistem peringkasan teks otomatis yang dapat mengambil inti dari isi artikel tanpa mengurangi makna dari artikel tersebut.

Peringkasan teks otomatis adalah proses meringkas untuk menghasilkan teks yang dipersingkat menggunakan komputer dengan tujuan menekankan informasi utama dan membantu pembaca memperoleh informasi utama dari teks tanpa perlu membaca seluruh isi teks atau dokumen. Peringkasan teks dapat dikategorikan menjadi dua pendekatan, yaitu ekstraktif dan abstraktif (Munot dan Govilkar, 2014). Peringkasan teks ekstraktif adalah peringkasan yang dilakukan dengan peringkasan dengan memilih kalimat-kalimat penting dari dokumen asli, sedangkan peringkasan teks abstraktif dilakukan dengan mengurangi isi dokumen teks secara semantis dan membutuhkan *Natural Language Processing* yang mendalam.

Pada penelitian ini akan dibangun sebuah sistem untuk melakukan peringkasan teks agar dapat membantu pembaca memperoleh inti dari sebuah informasi dari sebuah berita dalam waktu yang singkat dengan menggunakan metode BM25. Beberapa penelitian peringkasan teks

telah dilakukan dengan menggunakan berbagai metode. Salah satunya adalah penelitian yang dilakukan oleh Barrios dkk. (2016) yang membandingkan metode *Longest Common Subsequence (LCS)*, *Cosine Similarity*, BM25, dan BM25+ dalam peringkasan teks otomatis menggunakan algoritme *TextRank*, disimpulkan bahwa metode BM25 menunjukkan hasil terbaik. Dimana dengan menggunakan metode BM25 terjadi peningkatan akurasi sebesar 2,92% dengan nilai $\epsilon = 0,25$, daripada hanya menggunakan algoritme *TextRank* yang asli tanpa tambahan kinerja metode lain. Metode lainnya yaitu BM25+, *Cosine Similarity* dan LCS juga membuat peningkatan tapi tidak lebih besar dari metode BM25, yaitu masing-masing sebesar 2,60%, 2,54% dan 1,40% (Barrios et al., 2016).

Salah satu penelitian lainnya adalah penelitian yang dilakukan oleh Pinandhita (2013) yang melakukan penelitian peringkasan teks otomatis berbasis kata benda. Terdapat empat metode yang diterapkan pada sistem yaitu BM25, *cosine similarity*, *content overlap*, dan koefisien *dice*. Hasil yang didapatkan setelah 7 kali percobaan adalah rata-rata hasil perhitungan koefisien *dice* menggunakan *PageRank* atau tanpa judul untuk *cosine* sebesar 0,363, BM25 0,442, dan *content overlap* 0,396, sedangkan rata-rata hasil perhitungan koefisien *dice* menggunakan judul untuk *cosine* sebesar 0,327, BM25 0,327, dan *content overlap* 0,310. Sehingga dapat disimpulkan bahwa metode yang memiliki kinerja terbaik adalah metode BM25 pada percobaan tanpa menggunakan judul (Pinandhita, 2013).

Berdasarkan uraian di atas serta dari beberapa penelitian yang telah dilakukan sebelumnya, maka muncul gagasan untuk membuat sistem peringkasan teks otomatis dengan menggunakan metode BM25. Sistem ini akan menggunakan pendekatan peringkasan teks secara ekstraktif dengan membuat isi dokumen menjadi lebih ringkas dan singkat sehingga pembaca dapat mengambil inti dari sebuah informasi dengan cepat. Hal tersebut dilakukan dengan mengambil kalimat-kalimat penting yang mewakili inti dokumen atau teks asli, lalu membuang kalimat-kalimat yang tidak penting. Proses pengambilan kalimat dilakukan dengan menghitung bobot tiap kata dari hasil *preprocessing* dengan metode TF IDF yaitu berdasarkan frekuensi kemunculan kata. Kemudian menerapkan metode BM25 yang merupakan metode untuk proses pemeringkatan

kalimat. Lalu sistem akan menampilkan kalimat yang memiliki peringkat paling tinggi.

2. DASAR TEORI

2.1. Data yang Digunakan

Pengumpulan data merupakan tahap yang penting dalam proses perhitungan sistem. Data yang digunakan sebagai masukan pada penelitian ini adalah dokumen artikel berita kategori *entertainment* dari situs kompas.com sebanyak 20 artikel. Selain itu, dibutuhkan data *stoplist* yaitu data yang berisi kata-kata yang tidak penting yang akan digunakan dalam proses sistem. Sedangkan data yang digunakan dalam pengujian sistem adalah data hasil ringkasan sistem dan data ringkasan manual dari pakar yang isinya tidak lebih dari 50% jumlah kalimat pada dokumen artikel asli. Pakar merupakan guru bahasa Indonesia di SMK Negeri 11 Bandung.

2.2. Preprocessing

Preprocessing merupakan tahap awal yang perlu dilakukan dalam pemrosesan teks untuk membentuk struktur data yang baik. Tujuan dari *preprocessing* adalah untuk mendapatkan indeks *term* dari dokumen teks agar dapat diproses untuk tahap berikutnya yaitu pemrosesan dengan metode BM25. Langkah-langkah yang dilakukan dalam tahap *preprocessing* adalah *parsing*, *cleaning*, tokenisasi, *filtering* dan *stemming*.

2.2.1 Parsing

Parsing adalah tahapan untuk melakukan pemisahan kalimat dengan ciri adanya simbol titik, simbol tanda tanya, atau simbol tanda seru. Keluaran dari tahap ini adalah kumpulan kalimat dalam dokumen teks yang akan digunakan untuk tahap berikutnya.

2.2.2 Cleaning

Tahapan ini dapat dikatakan adalah tahap pembersihan *noise* dalam teks. *Noise* merupakan suatu bentuk data yang nantinya mungkin akan mengganggu proses pengolahan data tersebut. *Noise* yang dimaksud dalam penelitian ini adalah karakter selain huruf alfabet pada teks.

2.2.3 Tokenisasi

Tokenisasi adalah tahapan untuk memisah setiap kata pada semua kalimat yang dihasilkan

pada proses *parsing* menjadi potongan kata-kata tunggal. Kata dalam teks yang dimaksud adalah kata yang dipisah oleh karakter spasi. Pada tahap ini juga akan dilakukan proses untuk mengubah huruf dalam teks menjadi huruf kecil, atau disebut dengan proses *case folding*.

2.2.4 Filtering

Filtering adalah tahap menghapus *stopword*. *Stopword* merupakan kata-kata yang sering muncul dalam dokumen teks yang berupa kata umum tetapi dianggap tidak mempunyai makna.

2.2.4 Stemming

Stemming merupakan tahapan untuk mengubah kata yang merupakan hasil dari proses *filtering* menjadi kata dasar semuanya. Tahap ini dilakukan dengan menghapus semua imbuhan baik imbuhan pada awalan kata maupun pada akhiran kata. Keluaran dari tahap ini nantinya akan dijadikan *term*.

2.3. Metode TF IDF

Metode TF IDF digunakan untuk menghitung nilai bobot dari *term* dan kalimat yang nilainya paling tinggi untuk ditampilkan dalam ringkasan. TF (*Term Frequency*) adalah frekuensi dari munculnya sebuah *term* dalam dokumen. Perhitungan nilai TF dapat dilakukan dengan Persamaan (1).

$$TF = \begin{cases} 1 + \log_{10}(f_{t,d}), & f_{t,d} > 0 \\ 0, & f_{t,d} = 0 \end{cases} \quad (1)$$

Nilai $f_{t,d}$ pada Persamaan (1) adalah jumlah *term* t pada dokumen d . Sedangkan IDF (*Inverse Document Frequency*) adalah perhitungan bagaimana sebuah *term* didistribusikan pada koleksi dokumen. Perhitungan nilai IDF dapat dilakukan dengan Persamaan (2).

$$IDF = \log\left(\frac{N}{df}\right) \quad (2)$$

Nilai N pada Persamaan (2) merupakan banyaknya dokumen dalam koleksi, sedangkan nilai df adalah banyaknya dokumen yang mengandung *term*.

2.3. Perhitungan Metode BM25

BM25 adalah tahap untuk melakukan pemerinkatan kalimat dengan cara menjumlahkan nilai bobot kata pada tiap kalimat. Kalimat yang sudah dilakukan perhitungan bobot akan disusun sesuai dengan

bobotnya dari yang terbesar hingga yang terkecil. Metode BM25 mirip dengan metode *cosine similarity* yaitu menggunakan TF dan IDF untuk perhitungan bobot *term*. Perbedaannya, pada metode BM25 ditambahkan parameter bebas k_1 dan b . Dimana nilai optimal untuk parameter k_1 dan b adalah $k_1 = 1,2$ dan $b = 0,75$ (Manning dan Raghavan, 2009). Langkah-langkah pada metode BM25 adalah sebagai berikut:

1. Perhitungan panjang kalimat
Perhitungan panjang kalimat adalah tahapan untuk menghitung jumlah kata yang ada pada tiap kalimat.
2. Perhitungan rata-rata panjang kalimat
Tahap ini akan melakukan proses perhitungan rata-rata jumlah kata yang ada di kalimat dalam dokumen.
3. Perhitungan nilai BM25
Perhitungan nilai BM25 dapat dilakukan dengan Persamaan (3).

$$Score = \sum_{i=1}^n IDF(q_i) \frac{TF(t,d) \cdot (k_1+1)}{TF(t,d) + k_1 \cdot (1-b + b \cdot \frac{|D|}{avgdl})} \quad (3)$$

Deskripsi dari Persamaan (3):

$TF(t, d)$: *Term Frequency* pada dokumen d

$|D|$: Panjang dokumen D

$avgdl$: Rata-rata panjang dokumen

IDF : *Inverse Document Frequency*

Persamaan (3) akan menghasilkan nilai yang negatif apabila sebuah kata muncul di lebih dari setengah jumlah kalimat dalam dokumen (Mussina, Aubakirov dan Trigo, 2016). Oleh karena itu, Barrios membuat modifikasi sederhana untuk perhitungan nilai IDF agar terhindar dari hasil negatif (Barrios et al., 2016). Perhitungan dari IDF dapat dilihat pada Persamaan (4).

Jika nilai $n(q_i) > N/2$ maka perhitungan IDF dilakukan dengan Persamaan (4). Sedangkan jika nilai $n(q_i) \leq N/2$ maka perhitungan IDF dilakukan dengan Persamaan (5).

$$IDF = \log(N - n(q_i) + 0.5 - (-\log(n(q_i) + 0.5))) \quad (4)$$

$$IDF = \epsilon \times avgIDF \quad (5)$$

Deskripsi dari Persamaan (4) dan Persamaan (5):

N : Jumlah total kalimat dalam dokumen

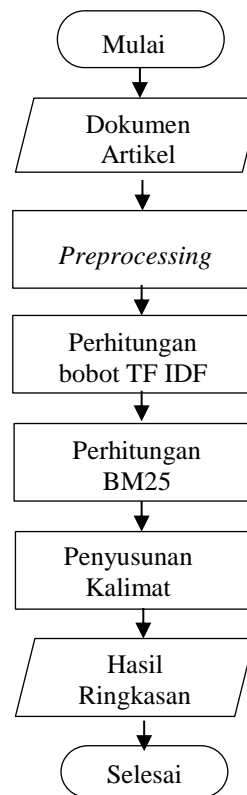
$n(q_i)$: Jumlah dokumen yang berisi *term* q_i

$avgIDF$: Rata-rata nilai IDF semua *term*

ϵ bernilai antara 0.3 dan 0.5, nilai ϵ yang digunakan adalah 0.5.

3. METODOLOGI

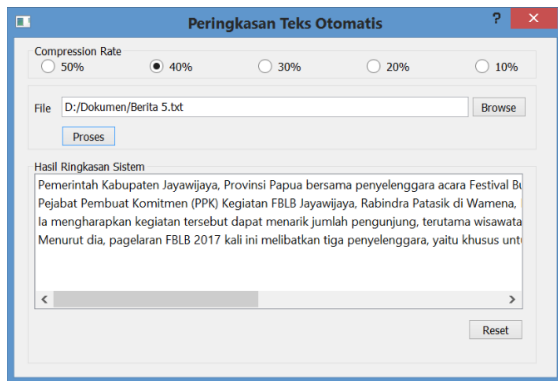
Tahapan pada implementasi sistem ditunjukkan pada Gambar 1.



Gambar 1. Alur Proses Algoritme

Dari alur *flowchart* pada Gambar 2, hal yang pertama kali dilakukan oleh sistem adalah dengan memasukkan masukan berupa dokumen artikel berita. Setelah itu dilakukan tahapan *preprocessing* seperti *parsing*, *cleaning*, tokenisasi, *filtering*, dan *stemming*. Kemudian melakukan perhitungan bobot TF dan IDF untuk mencari bobot dari setiap *term*. Setelah itu, melakukan proses perhitungan BM25 sesuai dengan Persamaan (3) untuk mendapatkan peringkat kalimat sesuai total jumlah bobotnya. Setelah mendapatkan peringkat untuk tiap kalimat, maka tahap selanjutnya adalah menyusun kalimat-kalimat sesuai dengan nilai tingkat kompresi yang dipilih oleh pengguna. Setelah itu, sistem akan mengeluarkan keluaran berupa hasil ringkasan dengan jumlah kalimat sesuai dengan tingkat kompresi yang pengguna inginkan.

Hasil dari implementasi peringkasan teks otomatis dengan menggunakan metode BM25 ditunjukkan pada Gambar 2.



Gambar 2. Hasil Implementasi

4. PENGUJIAN DAN ANALISIS

Ringkasan yang dihasilkan pada sistem peringkasan teks otomatis perlu diuji dan dievaluasi agar mengetahui akurasi dan kesesuaian hasil ringkasan yang dihasilkan. Evaluasi yang dilakukan pada penelitian ini adalah evaluasi intrinsik dengan membandingkan ringkasan yang dibuat oleh sistem dengan ringkasan manual yang dibuat oleh pakar. Adapun pengujian dilakukan dengan cara membandingkan nilai *precision*, *recall*, dan *f-measure* pada setiap nilai *compression rate*. Pada penelitian ini, nilai *compression rate* yang digunakan adalah sebesar 10%, 20%, 30%, 40%, dan 50% untuk setiap data artikel berita yang berbentuk teks. Nilai tersebut digunakan karena sebaiknya sebuah ringkasan tidak melebihi dari 50% keseluruhan isi dokumen sebenarnya (Alguliev dan Aliguliyev, 2009).

a. *Precision*

Presisi merupakan tingkat ketepatan dari peringkasan otomatis yang dilakukan sistem. Presisi adalah jumlah kalimat yang ada dalam sistem dan ditemukan juga dalam ringkasan manual dibagi dengan jumlah kalimat dalam ringkasan sistem.

b. *Recall*

Merupakan tingkat keberhasilan sistem dalam peringkasan otomatis yang menentukan berapa proporsi kalimat yang dipilih oleh manusia yang juga dipilih oleh sistem. *Recall* adalah jumlah kalimat yang ditemukan di kedua sistem dan ringkasan manual dibagi dengan jumlah kalimat pada ringkasan manual.

c. *F-Measure*

F-Measure adalah gabungan dari nilai *precision* dan *recall*. Perhitungan nilai *F-measure* ditunjukkan pada Persamaan (5).

$$F = \frac{1}{\frac{1}{2}(\frac{1}{P} + \frac{1}{R})} = \frac{2 \times P \times R}{P+R} \quad (5)$$

Dimana P adalah nilai dari *precision*, sedangkan R adalah nilai dari *recall*. *Precision*, *recall* dan *F-Measure* memiliki rentang nilai diantara 0 dan 1. Apabila nilai *precision*, *recall*, dan *f-measure* semakin tinggi, maka semakin baik pula hasil peringkasan otomatis yang dilakukan oleh sistem.

4.1. Pengujian *Compression Rate* 50%

Tabel 1. Pengujian *Compression Rate* 50%

Dokumen ke-	Pengujian		
	<i>Precision</i>	<i>Recall</i>	<i>FM</i>
1	0,583	0,700	0,636
2	0,769	0,909	0,833
3	0,833	0,833	0,833
4	0,600	0,667	0,632
5	0,600	0,750	0,667
6	0,750	0,750	0,750
7	0,833	0,833	0,833
8	0,625	0,833	0,714
9	0,667	0,750	0,706
10	0,583	0,583	0,583
...
...
19	0,722	0,722	0,722
20	0,857	0,857	0,857
Rata-rata	0,691	0,730	0,709

Dari hasil pengujian yang dapat dilihat pada Tabel 1, dapat diketahui hasil dari *precision*, *recall*, dan *f-measure* pada saat *compression rate* 50% dari seluruh dokumen. Nilai tertinggi dari *precision*, *recall*, dan *f-measure* didapatkan pada saat meringkas dokumen ke-20. Nilai *precision* akan semakin tinggi apabila jumlah kalimat yang ditemukan pada ringkasan sistem dan ditemukan juga pada ringkasan manual semakin banyak. Sedangkan nilai *recall* yang tinggi terjadi apabila proporsi antara jumlah kalimat pada ringkasan sistem dan ringkasan manual seimbang. Rata-rata nilai *f-measure* adalah sebesar 0,709.

4.2. Pengujian *Compression Rate* 40%

Tabel 2. Pengujian *Compression Rate* 40%

Dokumen ke-	Pengujian		
	<i>Precision</i>	<i>Recall</i>	<i>FM</i>
1	0,500	0,556	0,526
2	0,800	0,889	0,842

3	0,800	0,800	0,800
4	0,500	0,571	0,533
5	0,750	0,750	0,750
6	0,667	0,667	0,667
7	0,800	0,800	0,800
8	0,667	0,800	0,727
9	0,750	0,857	0,800
10	0,600	0,600	0,600
...
...
19	0,786	0,786	0,786
20	0,833	0,833	0,833
Rata-rata	0,711	0,734	0,722

Dari hasil pengujian yang dapat dilihat pada Tabel 2, dapat diketahui hasil dari precision, recall, dan f-measure pada saat compression rate 40% dari seluruh dokumen. Dimana terjadi peningkatan rata-rata nilai *precision* dan penurunan nilai *recall*. Peningkatan nilai *precision* dikarenakan berkurangnya jumlah kalimat dibanding saat *compression rate* 40%, dan kalimat yang dihasilkan sistem banyak yang ditemukan juga pada ringkasan manual. Sedangkan penurunan nilai *recall* disebabkan proporsi kalimat antara ringkasan manual dan ringkasan manual menurun keseimbangannya.

4.3. Pengujian Compression Rate 30%

Tabel 3. Pengujian Compression Rate 30%

Dokumen ke-	Penguujian		
	Precision	Recall	FM
1	0,500	0,571	0,533
2	0,625	0,714	0,667
3	1,000	1,000	1,000
4	0,500	0,500	0,500
5	1,000	1,000	1,000
6	0,667	0,667	0,667
7	1,000	1,000	1,000
8	0,800	0,800	0,800
9	0,833	0,833	0,833
10	0,714	0,714	0,714
...
...
19	0,636	0,636	0,636
20	0,800	0,800	0,800
Rata-rata	0,730	0,738	0,734

Pada Tabel 3 dapat diketahui bahwa rata-rata nilai dari precision, recall, dan f-measure saat nilai compression rate 30% secara berturut-turut adalah 0,730, 0,738, dan 0,734. Nilai ini meningkat dari nilai saat *compression rate* sebelumnya. Hal ini disebabkan kalimat yang dihasilkan sistem pada saat *compression rate*

30% banyak yang muncul juga pada ringkasan manual dari pakar sehingga nilai *precision* yang didapatkan lebih tinggi. Selain itu, proporsi antara jumlah kalimat pada ringkasan sistem dengan ringkasan manual juga seimbang. Nilai tertinggi yang didapatkan adalah 1 untuk masing-masing nilai *precision*, *recall*, dan *f-measure*. Hal ini menunjukkan sistem mempunyai ketepatan ringkasan dan keberhasilan yang tinggi karena kalimat yang dihasilkan sistem semuanya ada pada ringkasan manual, selain itu proporsi antara jumlah kalimat pada sistem dengan jumlah kalimat pada ringkasan manual juga seimbang.

4.4. Pengujian Compression Rate 20%

Tabel 4. Pengujian Compression Rate 20%

Dokumen ke-	Penguujian		
	Precision	Recall	FM
1	0,600	0,750	0,667
2	0,800	0,800	0,800
3	0,667	0,667	0,667
4	0,500	0,500	0,500
5	1,000	1,000	1,000
6	1,000	1,000	1,000
7	0,667	0,667	0,667
8	0,667	0,667	0,667
9	0,500	0,500	0,500
10	0,600	0,600	0,600
...
...
19	0,429	0,429	0,429
20	0,667	0,667	0,667
Rata-rata	0,649	0,656	0,652

Pada Tabel 4 dapat diketahui bahwa rata-rata nilai dari *precision*, *recall*, dan *f-measure* saat nilai *compression rate* 20% secara berturut-turut adalah 0,649, 0,656, dan 0,652. Nilai ini menunjukkan adanya penurunan nilai *precision*, *recall*, dan *f-measure* pada saat *compression rate* 20%. Hal ini dikarenakan jumlah kalimat yang dihasilkan sistem semakin sedikit. Selain itu kalimat pada ringkasan yang dihasilkan sistem belum tentu ada pada ringkasan manual dari pakar.

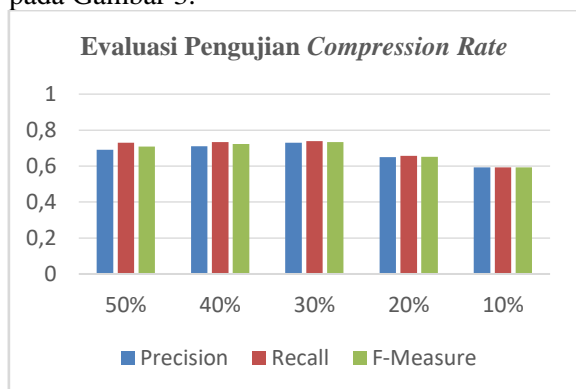
4.5. Pengujian Compression Rate 10%

Tabel 5. Pengujian Compression Rate 10%

Dokumen ke-	Penguujian		
	Precision	Recall	FM
1	0,667	0,667	0,667
2	1,000	1,000	1,000
3	0,500	0,500	0,500

4	1,000	1,000	1,000
5	0,000	0,000	0,000
6	1,000	1,000	1,000
7	0,500	0,500	0,500
8	0,500	0,500	0,500
9	0,500	0,500	0,500
10	0,333	0,333	0,333
...
...
19	0,500	0,500	0,500
20	0,500	0,500	0,500
Rata-rata	0,592	0,592	0,592

Pada Tabel 6.5 dapat diketahui bahwa rata-rata nilai dari *precision*, *recall*, dan *f-measure* saat nilai *compression rate* 10% adalah 0,592 untuk masing-masing nilai. Nilai terendah adalah pada saat meringkas dokumen ke-5, dimana nilai nya adalah 0 untuk *precision*, *recall*, dan *f-measure*. Hal ini berarti tidak ada satu pun kalimat pada ringkasan manual yang muncul pada ringkasan sistem. Keseluruhan pengujian nilai *compression rate* dapat dilihat pada Gambar 3.



Gambar 3. Hasil Pengujian Compression Rate

Pada Gambar 3 diketahui rata-rata nilai *precision*, *recall*, dan *f-measure* untuk masing-masing nilai *compression rate*. Rata-rata nilai tertinggi didapatkan saat *compression rate* sebesar 30%. Nilai *precision* akan semakin tinggi apabila jumlah kalimat yang ditemukan pada ringkasan sistem dan ringkasan manual tinggi. Sedangkan nilai *recall* akan semakin tinggi apabila proporsi antara jumlah kalimat pada ringkasan sistem dan ringkasan manual seimbang. Maka dari itu dapat disimpulkan saat *compression rate* 30%, proporsi jumlah kalimat antara ringkasan sistem dengan ringkasan manual dari pakar cukup seimbang, dan banyak kalimat pada hasil ringkasan sistem yang dapat ditemukan juga di ringkasan manual dari pakar. Sedangkan rata-rata nilai terendah adalah pada saat nilai *compression rate* sebesar 10%. Hal ini

dikarenakan, semakin rendah nilai *compression rate*, maka jumlah kalimat pada hasil ringkasan sistem akan semakin sedikit. Kalimat tersebut belum tentu merupakan kalimat yang ada pada ringkasan manual. Sehingga menghasilkan nilai *precision*, *recall*, dan *f-measure* yang rendah.

Evaluasi pengujian *compression rate* berguna untuk menentukan berapa jumlah kalimat yang paling baik untuk peringkasan teks. Pada hasil pengujian yang telah dilakukan, diketahui bahwa nilai *precision*, *recall*, dan *f-measure* yang paling tinggi didapatkan pada saat *compression rate* 30%, dimana nilainya secara berurutan adalah 0,730, 0,738, dan 0,734.

5. KESIMPULAN

Setelah melakukan implementasi dan pengujian serta analisis pada sistem, dapat diambil kesimpulan bahwa peringkasan teks otomatis pada artikel berita hiburan berbahasa Indonesia dapat diterapkan dengan menggunakan metode BM25 secara ekstraktif. Proses dari implementasi metode BM25 pada peringkasan teks otomatis pada artikel berita hiburan berbahasa Indonesia adalah pertama, melakukan proses *preprocessing* terhadap dokumen yang dipilih, lalu melakukan proses perhitungan bobot TF dan IDF. IDF yang digunakan adalah IDF pada rumus metode BM25, setelah itu melakukan perhitungan BM25 sehingga akan menghasilkan nilai bobot tiap kalimat dalam dokumen artikel berita hiburan, lalu menyusun kalimat berdasarkan bobot dari proses perhitungan BM25, dan menampilkan hasil ringkasan dengan jumlah kalimat sesuai dengan tingkat kompresi yang dipilih. Kalimat yang akan ditampilkan adalah kalimat yang mempunyai nilai bobot paling tinggi.

Pada pengujian *compression rate*, hasil ringkasan terbaik didapatkan saat nilai *compression rate* yang digunakan adalah 30%. Dimana nilai rata-rata *precision*, *recall*, dan *f-measure* secara berurutan adalah sebesar 0,730, 0,738, dan 0,734.

DAFTAR PUSTAKA

- Alguliev, R. dan Aliguliyev, R., 2009. Evolutionary Algorithm for Extractive Text Summarization. *Intelligent Information Management*.
- APJII, 2017. *Penetrasi & Perilaku Pengguna Internet Indonesia 2017*. [daring] Asosiasi Penyelenggara Jasa Internet Indonesia. Tersedia pada:

<<https://apjii.or.id/survei2017>>.

- Barrios, F., López, F., Argerich, L. dan Wachenchauser, R., 2016. Variations of the Similarity Function of TextRank for Automated Summarization. [daring] Tersedia pada: <<http://arxiv.org/abs/1602.03606>>.
- Manning, C.D. dan Raghavan, P., 2009. An Introduction to Information Retrieval. In: *Online*. [daring] hal.1. Tersedia pada: <<http://dspace.cusat.ac.in/dspace/handle/123456789/2538>>.
- Munot, N. dan Govilkar, S.S., 2014. Comparative Study of Text Summarization Methods. *International Journal of Computer Applications*, 102(12), hal.975–8887.
- Mussina, A., Aubakirov, S. dan Trigo, P., 2016. Automatic Document Summarization based on Statistical Information.
- Pinandhita, R.R., 2013. *Peringkasan Dokumen Berbahasa Indonesia Berbasis Kata Benda Dengan BM25*. INSTITUT PERTANIAN BOGOR.