Jurnal Elektronik Ilmu Komputer Udayana
Volume 8, No 2. November 2019

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Perbandingan Jenis TF terhadap Hasil Evaluasi Information Retrieval

I Putu Gede Hendra Suputra[1], Kiki Dwi Prebiana [2], Frisca Olivia Gorianto [3]

Program Studi Teknik Informatika, FMIPA, Universitas Udayana
Jimbaran, Bali, Indonesia
hendra.suputra@unud.ac.id[1],kikidwiprebiana@gmail.com[2], fgorianto@gmail.com[3]

Abstract

Pada sebuah sistem temu kembali,salah satu cara untuk mencari kesamaan antara query dengan dokumen adalah dengan menggunakan Term Frequency – Inverse Document Frequency atau TF-IDF. TF yang umum digunakan adalah langsung menggunakan jumlah term frequency padahal banyak jenis TF lainnya yang dapat dikombinasikan dengan IDF. Penelitian ini akan mengkombinasikan 4 jenis TF, yaitu Natural TF, Normalization/max TF, Logaritma TF, dan Boolean TF dengan tujuan untuk mencari jenis TF mana yang lebih baik setelah dikombinasikan dengan IDF. Hasil penelitian menunjukkan bahwa.Logaritma TF adalah yang terbaik dengan nilai F-measure sebesar 0,00662.
Keywords: *TF-IDF, Natural TF, Normalization TF, Logaritma TF, Boolean TF*

1. Pendahuluan

Sistem temu kembali informasi adalah proses pengembalian informasi yang relevan sesuai dengan kebutuhan pengguna. Sistem temu kembali informasi akan mengembalikan dokumen relevan yang tersimpan sesuai dengan query yang diinputkan oleh user. Secara umum dokumen ynag relevan ditampilkan secara berurutan dari dokumen yang memiliki tingkat relevan paling tinggi ke dokumen yang memiliki tingkat relevan yang paling rendah [1].

Salah satu proses dalam sistem temu kembali adalah pembobotan teks. Pada penelitian sebelumnya proses pemboboan teks dilakukan dengan menerapkan dua buah metode yaitu pembobotan dengan menggunakan TF-IDF dan LCS. Dari penelitian yang dilakukan menunjukan bahwa hasil presisi dan recall yang diperoleh dari kedua metode tersebut adalah sama [2]. Selain itu penelitian lain terkait pembobotan teks juga dilakukan untuk membandingkan pengaruh penggunaan Raw TF-IDF atau Natural TF-IDF dengan max TF-IDF atau Normalization TF-IDF. Dari penelitian yang dilakukan tersebut menunjukan bahwa penerapan max TF-IDF pada proses pembobotan teks selalu menghasilkan nilai yang lebih baik jika dibandingkan dengan penggunaan Raw TF-IDF. Selain membandingkan antara kedua rumus TF – IDF yang berbeda, pada penelitian tersebut juga menunjukan bahwa proses perhitungaan kedekatan dengan dokumen menggunkaan Sosin Similarity selalu lebih baik dibandingkan penggunaan dengan Euclidean Distance [3].

Sampai saat ini metode pembobotan teks yang paling sering digunakan adalah metode TF-IDF . Metode TF-IDF adalah merupakan cara untuk memeberikan bobot hubungan suatu kata terhadap dokumen. Pada metode ini proses perhitungan bobot teks dilakukan dengan menghitung frekuensi kemunculan kata dalam dokumen dan inferse frekuensi dari kata tersebut. Semakin tinggi frekuensi suatu teks terhadap suatu dokumen menunjukan bahwa hubungan kata terhadap suatu dokumen juga semakin tinggi. Proses perhitungan bobot teks dengan TF – IDF sendiri memiliki beberapa macam proses perhitungan, yaitu Boolean TF, Logaritma TF, Natural TF, dan normalisasi TF atau max TF.

Oleh karena itu, pada penelitian ini akan dilakukan proses membandingkan pengaruh penggunaan jenis – jenis TF – IDF yang ada terhadap hasil yang diperoleh pada sistem temu kembali informasi. Proses membandingkan pengaruh penggunaan TF – IDF akan dilakukan dengan membandingkan nilai presisi, recall, dan F-Measure yang diperoleh dari setiap jenis TF-IDF yang digunakan.

2. Metode Penelitian

2.1. Data

Pada penelitian ini data yang digunakan adalah data sekunder yang diperoleh dari koleksi data yang dimiliki University of Glasgow. Data yang digunakan adalah data Library and Information Science Abstracts (LISA) [4] merupakan dokumen yang berisi judul serta abstrak dari karya ilmiah. Jumlah seluruh dokumen yang digunakan adalah sebanyak ***950*** dokumen.

2.2. Preprocessing

Preprocessing merupkan tahap awal dari proses sistem temu kembali. Pada proses ini terbagi menjadi beberapa tahap yaitu stopword removal, stemming, tokenization, dan term weighting.

2.2.1. Stopword Removal

Proses menentukan kata – kata penting dari hasil tokenisasi.

2.2.2. Tokenization

Proses memecah dokumen menjadi kumpulan kata. Proses ini dapat dilakukan dengan menghilangkan tanda baca atau memisahkannnya berdasarkan spasi.

2.2.3. Stemming

Proses merubah kata yang ada pada dokumen menjadi kata dasar. Hal ini dilakukan dengan cara menghilangkan imbuhan yang ada pada kumpulan kata yang ada pada dokumen.

2.2.4. Term Weight

Proses memberikan nilai terhadap kata yang ada pada dokumen. Pada proses ini perhitungan bobot akan dilakukan dengan menerapkan metode TF-IDF [5]. Pada algoritma TF-IDF rumus yang digunakan untuk menghitung bobot w untuk masing – masing kata pada suatu dokumen adalah:

$$W_{dt} = TF_{dt} * IDF_t \quad (1)$$

$$IDF_t = \log (N/df) \quad (2)$$

Dimana:

W_{dt} = Bobot kata ke t dalam dokumen d
TF_{dt} = Frekuensi suatu term dalam dokumen
IDF_t = Inverse Frekuensi suatu dokumen
N = total Dokumen
Df = Banyaknya dokuemn yang mengandung term tersebut

2.3. Term- Frequency – Inverse Document Frequency

Metode term Frequency – Inverse Document Frequency atau TF-IDF adalah sebuah metode yang terkenal efisien, mudah dan memiliki hasil yang akurat digunakan pada information retrival untuk menghitung bobot term [6]. Metode TF-IDF adalah cara pemberian bobot suatu kata pada dokumen. TF-IDF digunakan untuk mengevaluasi tentang seberapa penting term tersebut pada suatu dokumen. Terdapat beberapa jenis perhitungan TF yang dapat digunakan. Pada penelitian ini akan digunakan empat jenis perhitungan TF yaitu Natural TF, Normalization TF atau max-TF, Logaritma TF, dan Boolean TF. Pada Natural TF maka nilai TF yang digunakan adalah nilai TF sebenarnya. Yaitu banyaknya kata tersebut alam suatu dokumen. Sedangkan pada ketiga jenis TF yang lain proses perhitungan nilai TF mengikuti persamaan berikut:

- Normalization TF / Max TF:

p-ISSN: 2301-5373
e-ISSN: 2654-5101

$$\text{Max TF: } 0.4 + \frac{\mathbf{0.4 \times TF}}{\mathbf{max(TF)}} \quad (3)$$

- Logaritma TF:
  $$\text{Log TF: } 1 + \log(TF) \quad (4)$$

- Boolean TF:
  $$\begin{cases} \mathbf{1 if\ TF > 0} \\ \mathbf{0 if\ TF < 0} \end{cases} \quad (5)$$

2.4. Skenario Pengujian

Proses pengujian dilakukan dengan menginputkan 20 query yang telah disediakan oleh dataset. Kemudian hasil yang diperoleh dari sistem akan dicocokan dengan hasil dokumen relevan untuk query tersebut. Pada pengujian ini akan dibandingkan hasil nilai presisi, recall, dan f-measure dari masing masing jenis term frequency (TF) yang digunakan.

3. Hasil dan Pembahasan

Tahap pertama yang perlu dilakukan adalah mem-preprocessing data yang akan dimasukkan ke dalam database. Pada tahap ini juga dilakukan perhitungan term frequency atau TF, nilai document frequency atau DF, dan nilai Inverse Document Frequency atau IDF. Setelah semua term sudah memiliki ketiga nilai tersebut maka tahap selanjutnya adalah masuk ke mesin pencari.

Query yang diinputkan akan dilakukan tahap preprocessing. Kemudian dilakukan perhitungan kesamaan antara query dengan dokumen yang ada di database menggunakan cosine similarity.

Tabel 1. Nilai Presisi, Recall, dan F-measure

Query	Natural TF	Normalization TF	Boolean TF	Logaritma TF
1	P: 0 R: 0 F: 0	P: 0 R: 0 F: 0	P: 0 R: 0 F: 0	P: 0 R: 0 F: 0
2	P: 0,047619048 R: 0,333333333 F: 0,083333333	P: 0,004103967 R: 1 F: 0,008174387	P: 0,004103967 R: 1 F: 0,008174387	P: 0,004103967 R: 1 F: 0,008174387
3	P: 0 R: 0 F: 0	P: 0,005535055 R: 1 F: 0,011009174	P: 0,005535055 R: 1 F: 0,011009174	P: 0,005535055 R: 1 F: 0,011009174
4	P: 0,015037594 R: 0,5 F: 0,02919708	P: 0,006031363 R: 0,833333333 F: 0,011976048	P: 0,007237636 R: 1 F: 0,014371257	P: 0,007237636 R: 1 F: 0,014371257
5	P: 0,010638298 R: 0,333333333 F: 0,020618557	P: 0,003911343 R: 1 F: 0,007792208	P: 0,0078125 R: 1 F: 0,015503876	P: 0,007822686 R: 1 F: 0,015523933
6	P: 0 R: 0 F: 0	P: 0 R: 0 F: 0	P: 0 R: 0 F: 0	P: 0 R: 0 F: 0
7	P: 0	P: 0,001254705	P: 0,001254705	P: 0,001254705

	R: 0 F: 0	R: 1 F: 0,002506266	R: 1 F: 0,002506266	R: 1 F: 0,002506266
8	P: 0 R: 0 F: 0	P: 0,003488372 R: 0,75 F: 0,006944444	P: 0,003488372 R: 0,75 F: 0,006944444	P: 0,003488372 R: 0,75 F: 0,006944444
9	P: 0,008403361 R: 0,333333333 F: 0,016393443	P: 0,001156069 R: 0,333333333 F: 0,002304147	P: 0,001769912 R: 0,333333333 F: 0,003521127	P: 0,001769912 R: 0,333333333 F: 0,003521127
10	P: 0 R: 0 F: 0	P: 0,001150748 R: 1 F: 0,002298851	P: 0,001150748 R: 1 F: 0,002298851	P: 0,001150748 R: 1 F: 0,002298851
11	P: 0 R: 0 F: 0	P: 0,001412429 R: 1 F: 0,002820874	P: 0,001412429 R: 1 F: 0,002820874	P: 0,001412429 R: 1 F: 0,002820874
12	P: 0,01754386 R: 0,111111111 F: 0,03030303	P: 0,00952381 R: 0,666666667 F: 0,018779343	P: 0,00952381 R: 0,666666667 F: 0,018779343	P: 0,00952381 R: 0,666666667 F: 0,018779343
13	P: 0,027777778 R: 0,75 F: 0,053571429	P: 0,005208333 R: 1 F: 0,010362694	P: 0,005208333 R: 1 F: 0,010362694	P: 0,005208333 R: 1 F: 0,010362694
14	P: 0,006622517 R: 0,5 F: 0,013071895	P: 0,001663894 R: 0,5 F: 0,00331675	P: 0,001663894 R: 0,5 F: 0,00331675	P: 0,001663894 R: 0,5 F: 0,00331675
15	P: 0 R: 0 F: 0	P: 0,002699055 R: 1 F: 0,00538358	P: 0,002702703 R: 1 F: 0,005390836	P: 0,002702703 R: 1 F: 0,005390836
16	P: 0 R: 0 F: 0	P: 0,001410437 R: 1 F: 0,002816901	P: 0,001410437 R: 1 F: 0,002816901	P: 0,001410437 R: 1 F: 0,002816901
17	P: 0 R: 0 F: 0	P: 0 R: 0 F: 0	P: 0 R: 0 F: 0	P: 0 R: 0 F: 0
18	P: 0,035714286 R: 0,555555556 F: 0,067114094	P: 0,009247028 R: 0,777777778 F: 0,018276762	P: 0,009259259 R: 0,777777778 F: 0,018300654	P: 0,009259259 R: 0,777777778 F: 0,018300654
19	P: 0 R: 0 F: 0	P: 0,001550388 R: 0,5 F: 0,00309119	P: 0,001550388 R: 0,5 F: 0,00309119	P: 0,001550388 R: 0,5 F: 0,00309119
20	P: 0 R: 0	P: 0,001557632 R: 1	P: 0 R: 0	P: 0,001557632 R: 1

p-ISSN: 2301-5373
e-ISSN: 2654-5101

	F: 0	F: 0,00311042	F: 0	F: 0,00311042

Tabel 1 menunjukkan hasil uji coba 20 query menggunakan 4 jenis TF yang berbeda. P adalah nilai presisi, R adalah recall, dan F adalah nilai F-measure. Nilai P dan R adalah 0 karena sistem tidak memunculkan dokumen-dokumen yang relevan sehingga.

4. Kesimpulan

Dari 20 percobaan yang telah dilakukan, dari Tabel 1 menunjukan bahwa pada penggunaan Normalization TF dan Logaritma TF memiliki hasil nilai recall yang lebih baik dibandingkan pada kedua jenis TF yang lain. Sedangkan pada nilai presisi kedua TF yaitu Normalization dan Logaritma menunjukan bahwa nilai Logaritma TF menunjukan nilai presisi yang lebih baik jika dibandingkan pada Normalization TF. Dari kedua nilai tersebut, pada Logaritma TF menunjukan nilai rata rata F-measure yang lebih baik dibandingkan pada penggunaan TF yang lain. Rata rata nilai F-measure pada Logaritma TF adalah sebesar 0,00662.

Sehingga dapat disimpulkan bahwa, penggunaan TF yang berbeda pada proses information retrieval mempengaruhi hasil dari information retrieval, dan pada penelitian ini hasil terbaik diperoleh dengan menggunakan Logaritma TF.

Akan tetapi pada penelitian ini masih memiliki beberapa kekurangan, diantaranya adalah lamanya waktu yang diperlukan untuk tahap preprocessing data. Dari seluruh data yang ada, pada penelitian ini peneliti hanya menggunakan 950 dokumen dari 6004 keseluruhan data yang ada. Diharapkan pada peneliatan selanjutnya, dapat dilakukan proses preprocessing yang lebih tepat sehingga seluruh data yang ada dapat digunakan dalam proses information retrieval sehingga pengaruh penggunaan TF pada proses ini dapat semakin jelas selain itu, diharapkan dapat meningkatkan nilai F-measure yang diperoleh.

Daftar Pustaka

[1] C. J. Van Rijsbergen, "Information retrieval," 1979.
[2] M. N. Saadah, R. W. Atmagi, D. S. Rahayu, and A. Z. J. J. J. I. T. I. Arifin, "Sistem Temu Kembali Dokumen Teks Dengan Pembobotan TF-IDF dan LCS," vol. 11, no. 1, pp. 19-22, 2013.
[3] B. K. Hananto, A. Pinandito, and A. P. J. J. P. T. I. d. I. K. e.-I. Kharisma, "Penerapan Maximum TF-Idf Normalization Terhadap Metode Knn Untuk Klasifikasi Dataset Multiclass Panichella Pada Review Aplikasi Mobile," vol. 2548, p. 964X, 2018.
[4] U. o. Glasgow. (1994). *IR Test Collections*. Available: http://ir.dcs.gla.ac.uk/resources/test_collections/lisa/
[5] C. Manning, P. Raghavan, and H. Schutze, "Introduction to Information Retrieval."
[6] A. A. Maarif, "Penerapan Algoritma TF-IDF Untuk Pencarian Karya Ilmiah," 2015.