Fuzzy Information Interval

Oleh :
Alvin Limassa 13516039
Jeffry 13516156

Information Retrieval (Temu Balik Informasi)

Temu balik informasi (*information retrieval*) adalah ilmu yang mempelajari tentang proses dan metode dalam pencarian dan pengambilan informasi yang dibutuhkan oleh pengguna (*query*) dari koleksi berbagai dokumen. Contoh temu balik informasi yakni berupa pencarian suatu informasi tertentu dari suatu dokumen, pencarian suatu dokumen tertentu, pencarian metadata yang menjelaskan data lain, maupun database yang berisi teks, gambar, maupun suara.

Temu balik informasi memiliki beberapa peran bagi pengguna yakni:

- Menganalisis isi sumber informasi dan pertanyaan pengguna (*query*)
- Mencocokkan pertanyaan pengguna (*query*) dengan sumber informasi untuk mendapatkan dokumen yang relevan, sehingga informasi yang diperlukan oleh pengguna dapat ditemukan.

Secara prinsip, temu balik informasi adalah suatu proses yang sederhana. Misalkan terdapat sumber informasi berupa kumpulan dokumen, dan suatu pertanyaan pengguna (*query*). Untuk memperoleh informasi yang diperlukan pengguna, dapat dilakukan dengan membaca semua dokumen yang ada, lalu menyimpan dokumen yang relevan dan mengesampingkan dokumen lainnya. Proses ini disebut *perfect retrieval*. Namun, proses ini tidaklah praktis sebab bila sumber informasi yang ada berjumlah banyak, maka pengguna akan menghabiskan banyak waktu untuk membaca semua dokumen yang tersedia. Oleh karena itu, maka dibuatlah berbagai model representasi sumber informasi agar dapat melakukan pengambilan informasi yang lebih efektif. Berbagai model tersebut antara lain :

Model Klasik :

- Model *set-theoritic*, dimana pada model ini, sumber informasi direpresentasikan sebagai kumpulan dari berbagai kata dan frasa. Model ini dapat dipecah menjadi dua model yang lebih spesifik yaitu model *fuzzy* dan *extended boolean*
- Model *algebraic*, dimana pada model ini, sumber informasi direpresentasikan sebagai matriks, *tuple*, atau vektor. Model ini dapat dipecah menjadi beberapa model yang lebih spesifik yaitu model *Generalized Vector, Latent Semantic, Indexing, Neural Networks.*
- Model *probabilistic*, dimana pada model ini, proses temu balik informasi dianggap sebagai inferensi probabilitas. Model ini dapat dipecah menjadi

beberapa model yang lebih spesifik yaitu model *Inference Network, Belief Network, Hidden Markov, Probabilistic LSI,* dan *Language*

Model Terstruktur :

- Model *non overlapping list,* dimana pada model ini membagi seluruh teks menjadi beberapa bagian yang tidak saling overlapping yang disimpan pada suatu list
- Model *Proximal nodes,* dimana pada model ini dilakukan pembagian indeks hirarki pada teks.

Perhitungan Bobot dengan TF-IDF

Dalam beberapa model temu balik informasi, pengambilan suatu informasi dari suatu dokumen akan melibatkan perhitungan bobot suatu *term* (bisa dalam bentuk kata, frasa, maupun unit hasil *indexing*). Salah satu cara perhitungan bobot *term* yakni dengan menggunakan TF-IDF. TF-IDF merupakan singkatan dari Term Frequency - Inverse Document Frequency.

TF (Term Frequency) adalah nilai yang merepresentasikan frekuensi dari kemunculan *term* yang bersangkutan dalam suatu dokumen. Karena itu, semakin banyak kemunculan suatu *term* pada suatu dokumen, maka akan berdampak semakin tinggi nilai TF dari dokumen tersebut. Untuk menghitung TF, dapat digunakan beberapa rumus, antara lain:

- TF biner, dimana TF dihitung berdasarkan ada atau tidaknya suatu *term* dalam dokumen tersebut, bila ada maka akan bernilai 1, dan bila tidak ada maka 0.
- TF murni, dimana TF dihitung berdasarkan frekuensi kemunculan suatu *term* dalam dokumen tersebut. Bila muncul 10 kali, maka TF akan bernilai 10.
- TF algoritmik, dimana TF akan 0 bila tidak ada *term,* dan bila ada, dihitung dengan rumus 1 + log10 (frekuensi kemunculan). Rumus ini digunakan untuk menghindari bias nilai TF untuk dokumen yang berisi sedikit *term* dari *query*, namun memiliki frekuensi kemunculan yang tinggi.
- TF normalisasi, dimana TF dihitung dengan perbandingan anatara frekuensi sebuah *term* dengan nilai maksimum dari keseluruhan atau kumpulan frekuensi *term* yang ada pada suatu dokumen.

IDF (Inverse Document Frequency) adalah nilai yang mencerminkan bagaimana suatu *term* didistribusikan secara luas pada koleksi dokumen yang ada. Semakin sedikit jumlah dokumen yang mengandung *term* yang bersangkutan, maka nilai IDF akan semakin besar. Jadi, rumus untuk menghitung IDF adalah log(D/fd), dimana D adalah jumlah dokumen dalam suatu koleksi, dan fd adalah jumlah dokumen yang mengandung suatu *term*.

Logika Fuzzy dan Fuzzy Set Theory

Logika *fuzzy* adalah logika yang memiliki nilai kebenaran di antara 2 nilai kebenaran tradisional, *true* dan *false*. Logika ini dibuat agar dapat mencakup kondisi dimana suatu nilai kebenaran bukanlah termasuk ke dalam *true* maupun *false.* Contoh cukup keras, dan sedikit dingin.

Berbeda dengan teori himpunan biasa, dimana suatu elemen dinyatakan terdapat dalam himpunan atau tidak, *fuzzy set theory* (teori himpunan *fuzzy*) memungkinkan derajat keanggotaan untuk suatu elemen dalam himpunan, sehingga memungkinkan peralihan keanggotaan yang lebih bertahap dibandingkan teori himpunan biasa.

Temu Balik Informasi dengan Metode Fuzzy

Salah satu jenis model *set-theoritic* yakni temu balik informasi yang menggunakan logika *fuzzy*. Metode *Fuzzy* adalah suatu metode yang merupakan pengembangan dari model boolean dan *fuzzy set theory.* Metode Fuzzy sendiri dapat dibagi menjadi dua metode klasik yaitu *Mixed Min and Max (MMM)* dan *Paice Model.* Kedua metode ini tidak dapat menghitung jumlah *query* yang digunakan akan tetapi alogritma ini masih termasuk ke dalam bagian algoritma P.

Pada pemodelan *Mixed Min and Max (MMM),* sebuah elemen dapat memiliki tingkat keanggotaan yang bervariasi (asumsikan d_A memetakan keanggotaan ke A), jumlah dokumen yang berhubungan dengan istilah pada indeks A dianggap sebagai tingkat keanggotaan dokumen dalam himpunan *fuzzy* yang terkait dengan A. Tingkat keanggotaan dapat diperoleh dengan menggunakan rumus berikut :

$$d_{A \cap B} = min(d_A, d_B)$$
$$d_{A \cup B} = max(d_A, d_B)$$

Berdasarkan contoh di atas, dokumen yang akan diambil untuk suatu *query* yang berbentuk A atau B harus memiliki asosiasi himpunan *fuzzy* dengan bentuk A gabungan B. Hal yang sama dilakukan untuk *query* dengan bentuk A dan B harus memiliki himpunan *fuzzy* dengan bentuk A irisan dengan B. Memanfaatkan kedua data tersebut, dimungkinkan untuk menentukan kemiripan suatu dokumen dengan dokumen yang lain menjadi $\min(d_A,d_B)$ atau $\max(d_A,d_B)$. Pemodelan jenis MMM mencoba untuk membandingkan kedua dokumen menggunakan operator boolean dengan mempertimbangkan kesamaan dokumen menjadi kombinasi linear dan nilai maks dan min yang di hasilkan.

Untuk memahami lebih lanjut tentang penggunakan nilai maks dan min, diberikan contoh sebagai berikut. Diberikan suatu dokumen D dengan bobot indeks $d_{A1},d_{A2},d_{A3},...,d_{An}$ untuk persyaratan $A_1,A_2,A_3,...,A_n$ memiliki query Q_{or}= (A_1 or A_2 or A_3 or … or A_n) dan Q_{and}= (A_1 and A_2 and A_3 and … and A_n). Pada pemodelan MMM query dokumen tersebut dimodelkan menjadi

$SIM(Q_{or}, D) = C_{or1} * max(d_{A1}, d_{A2}, ..., d_{An}) + C_{or2} * min(d_{A1}, d_{A2}, ..., d_{An})$

$SIM(Q_{and}, D) = C_{and1} * min(d_{A1}, d_{A2}, ..., d_{An}) + C_{and2} * max(d_{A1}, d_{A2}..., d_{An})$ dimana $C_{or1\ dan}\ C_{or2}$ merupakan nilai koefisien yang lebih lembut (persyaratan yang lebih tidak ketat), untuk memaksimalkan hasil *query* yang didapatkan maka pada umumnya akan memiliki nilai $C_{or1} > C_{or2}$ dan $C_{and1} > C_{and2}$ yang dapat disederhanakan menjadi $C_{or1} = 1 - C_{or2}$ dan $C_{and1} = 1 - C_{and2}$. Hasil percobaan membuktikan bahwa performansi optimal C_{and1} akan muncul pada rentang 0,5 - 0,8 dan C_{or1} >0,2. Secara umum metode MMM memiliki biaya komputasi yang rendah dan tingkat temu balik informasi yang jauh lebih baik dari model *boolean* standar.

Model *Paice* merupakan model yang mirip dengan model MMM, namun selain mempertimbangkan tingkat keanggotaan, model *Paice* juga mempertimbangkan bobot dari semua *term* dalam suatu dokumen dalam menentukan kemiripan suatu dokumen. Rumus untuk menghitung kemiripan suatu dokumen pada model *Paice* dapat ditulis sebagai berikut.

$$S(D,Q) = \sum_{i=1}^{n} \frac{r^{i-1} * w_{di}}{\sum_{j=1}^{n} r^{j-1}}$$

dimana D merupakan hasil TF-IDF dari dokumen tersebut, Q merupakan *query,* n merupakan banyaknya *query* yang diberikan, r merupakan konstanta, dan wdi merupakan bobot *term* ke-i dari dokumen. Ketika n=2, model *Paice* akan mirip dengan model MMM. Untuk *query* “dan”, r yang cocok sebesar 1,0 , sedangkan untuk *query* “atau”, r yang cocok sebesar 0,7.

Pembagian Tugas

Alvin Limassa : Mencari sumber informasi, melengkapi artikel, membuat GitHub
Jeffry : Merangkum sumber informasi, merapikan artikel

Referensi

1. Colvin, E. & Kraft, D. (2015). *Fuzzy Retrieval for Software Reuse*. Journal of the Association for Information Science and Technology Volume 67 Issue 10 October 2016, 2454-2463
2. Zadeh, L. A. et al. (1996). *Fuzzy Sets, Fuzzy Logic, Fuzzy Systems*, World Scientific Press
3. Singh, P., Dhawan, S., Agarwal, S., & Thakur, N. (2015). *Implementation of an efficient Fuzzy Logic based Information Retrieval System*. EAI Endorsed Trans. Scalable Information Systems, 2, e5.
4. https://ligiaprapta17.wordpress.com/2015/03/03/pengertian-information-retrieval-ir-peranan-ir-dan-contoh-contoh-ir/ , diakses pada 28 Mei 2018 pukul 13.20
5. https://dimas347.wordpress.com/2009/06/28/information-retrieval-system/ , diakses pada 28 Mei 2018 pukul 14.02
6. https://en.wikipedia.org/wiki/Information_retrieval , diakses pada 28 Mei 2018 pukul 11.20
7. https://pdfs.semanticscholar.org/presentation/d22a/67fb07b924230e5223e7938bb29d25f89d98.pdf , diakses pada 29 Mei 2018 pukul 6.50
8. https://informatikalogi.com/term-weighting-tf-idf/ , diakses pada 28 Mei 2018 pukul 19.50