OPTIMASI HASIL PENCARIAN PADA WEB SCRAPPING MENGGUNAKAN PEMBOBOTAN KATA TF-IDF

Oleh
Edy Prayitno[1)], Totok Suprawoto[2)], Beny Fajar Riyanto[3)]
[1,2,3]Program Studi Sistem Informasi, STMIK AKAKOM Yogyakarta
Jl Raya Janti No 143 Karangjambe Yogyakarta,
Email: [1]edyprayitno@akakom.ac.id, [2]totok@akakom.ac.id

Abstrack

This research is motivated by the amount of information offered on various webs. The large number of relevant websites related to the information sought causes users to have to search for the desired information one by one on the web so that the time required becomes longer. This research uses web scraping and TF-IDF method. Web scraping is a technique for getting information from web pages. In doing scraping, curl and simple html dom are needed to parse the scraped data. TF-IDF is a method to perform a search by looking for similarity of data with the keywords entered so that by using TF-IDF it is hoped that information that is more in line with the keywords entered is obtained. By using web scraping, additional data can be added to the system without using a web service. The use of TF-IDF results in a better search because the search is done by comparing the similarity of words between the data in the system and the search keywords.

Keywords: Curl, Simple Html Dom, Search Information System, TF-IDF, Web Scraping

PENDAHULUAN

Informasi lowongan kerja saat ini dapat diakses di berbagai media, seperti koran, brosur, atau papan pengumuman di perusahaan terkait, bahkan dapat melalui *website* milik pihak ketiga. Pihak ketiga akan menyediakan informasi lowongan dari berbagai perusahaan di dalam *website* milik mereka, contohnya jobs.id, job.id loker.id, dan lain-lain. Adanya pihak ketiga ini sangat membantu para pencari kerja untuk mendapatkan informasi lowongan kerja. Namun terkadang informasi lowongan kerja dari satu *website* dengan *website* lainnya berbeda, karena perusahan pemilik lowongan tidak memberikan semua informasi lowongan kerja kepada pihak ketiga Hal ini membuat para pencari kerja harus mengunjungi beberapa *website* untuk mendapatkan informasi lowongan yang sesuai untuk mereka. Akibatnya diperlukan waktu yang lebih untuk membuka *website* dan melakukan pencarian lowongan dari satu *website* ke *website* lainnya. Selain itu dalam mencari lowongan, beberapa website tidak mendukung pencarian dengan kata kunci lebih dari satu.

Untuk membantu pencari kerja dalam mencari lowongan tanpa harus mengunjungi satu per satu *website* penyedia lowongan kerja, maka dibutuhkan sebuah sistem yang dapat menampilkan informasi lowongan dari beberapa *website* lowongan kerja serta dapat melakukan pencarian lowongan kerja dengan menggunakan kata kunci lebih dari satu. Untuk melakukan pencarian dengan banyak kata kunci dapat menggunakan teknik pembobotan kata. Salah satu teknik pembobotan kata adalah TF-IDF (*term frequency-inverse document frequency)*.

Pengelompokkan kata dengan menggunakan TF-IDF dapat memperoleh akurasi sebesar 94.67% [1]. Akurasi tersebut diperoleh dari perbandingan hasil pengelompokan algoritma Naive Bayes yang sudah dikategorikan dengan data yang dikelompokkan secara manual menggunakan TF-IDF. Tingkat akurasi juga dipengaruhi oleh jumlah data. Semakin banyak jumlah data uji

maka semakin tinggi tingkat akurasi yang dihasilkan, bahkan bisa mencapai 100%. Hal ini dapat terjadi karena hasil pada kategori data hasil sistem sesuai dengan data asli [2].

LANDASAN TEORI

Tf-idf

Tf-Idf merupakan salah satu metode popular yang digunakan dalam menentukan bobot setiap kata. Metode Tf-Idf adalah cara pemberian bobot hubungan suatu kata terhadap dokumen [3]. Bobot tersebut dapat mencerminkan tingkat kepentingannya dalam sebuah dokumen. Bobot setiap kata akan dipetakan dalam lingkungan vektor, sehingga akan terbentuk lingkungan vektor berdimensi n [4]. TF-IDF merupakan salah satu metode terpopuler yang banyak digunakan untuk karakterisasi teks berbasis VSM (*Vector Space Model*) [5]. Bobot TF-IDF dapat diartikan sebagai keputusan relevansi yang merupakan hasil sederhana dari model keputusan probabilistik. Model membedakan dua jenis relevansi; satu jenis umum adalah relevansi dokumen lengkap yang berlaku untuk semua dokumen, dan jenis baru adalah relevansi parsial yang hanya berlaku untuk lokasi dokumen tertentu. Model membuat keputusan relevansi lokal untuk setiap lokasi dokumen, dan menggabungkan keputusan relevansi lokal ini ke dalam keputusan nrelevansi seluruh dokumen dari keseluruhan dokumen [6]. Dengan kata lain tingkat relevansi meningkat secara proporsional dengan berapa kali sebuah kata muncul dalam dokumen dan diimbangi oleh frekuensi kata dalam korpus. Frekuensi kata dihitung menggunakan persamaan (1) dimana frekuensi kata ke-i merupakan frekuensi kemunculan kata ke-i pada dokumen ke-j. Frekuensi dokumen terbalik adalah logaritma dari rasio jumlah dokumen dalam korpus dengan jumlah dokumen yang ditulis secara matematis seperti pada persamaan (2). Nilai yang diperoleh dengan mengalikan keduanya dirumuskan dalam persamaan (3) [7].

$$tf(i) = \frac{freq_i(d_j)}{\sum_{i=1}^{k} freq_i(d_j)}, \quad (1)$$

$$idf_i = \log \frac{|D|}{|\{d : t_i \in d\}|} \quad (2)$$

$$(tf - idf)_{ij} = tf_i(d_i).idf_i \quad (3)$$

Web-scraping

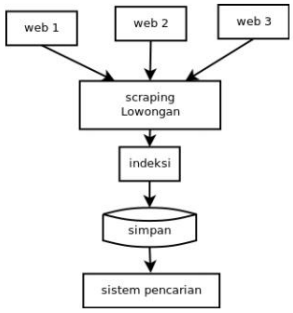
Web scraping merupakan proses pengambilan sebuah dokumen semi terstruktur dari internet, umumnya berupa halaman-halaman web dalam bahasa markup seperti HTML atau XHTML dan menganalisis dokumen tersebut untuk diambil data tertentu dari halaman tersebut untuk digunakan bagi kepentingan lain [8]. Scraping web, juga dikenal sebagai ekstraksi web atau pemanenan, adalah teknik untuk mengekstraksi data dari World Wide Web (WWW) dan menyimpannya ke sistem file atau *database* untuk pengambilan atau analisis selanjutnya. Umumnya, data web dihapus menggunakan Hyper-text Transfer Protocol (HTTP) atau melalui browser web. Ini dilakukan secara manual oleh pengguna atau secara otomatis oleh bot atau perayap web. Proses scraping data dari internet dapat dibagi menjadi dua langkah berurutan; memperoleh sumber daya web dan kemudian mengekstrak informasi yang diinginkan dari data yang diperoleh. Secara khusus, program penggalian web dimulai dengan menyusun permintaan HTTP untuk mendapatkan sumber daya dari situs web yang ditargetkan. Permintaan ini dapat diformat dalam URL yang berisi permintaan GET atau bagian dari pesan HTTP yang berisi *query* POST. Setelah permintaan berhasil diterima dan diproses oleh situs web yang ditargetkan, sumber daya yang diminta akan diambil dari situs web dan kemudian dikirim kembali ke program pengikisan web [9].

MEMETODE PENELITIAN

Penelitian ini menggunakan data lowongan kerja yang bersumber dari loker.id, jobs.is, dan job.id. Alur proses yang dilakukan dalam penelitian ini dapat dilihat pada Gambar 1.

ISSN 2798-3471 (Cetak)
ISSN 2798-3641 (Online)

Gambar 1. Alur proses

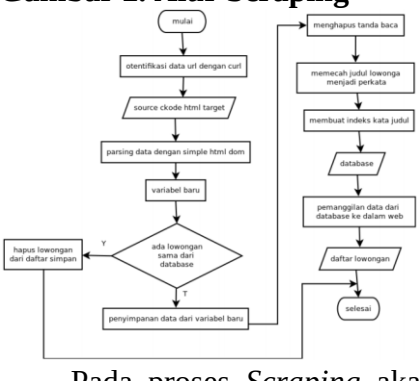


Proses pertama adalah proses *scraping* lowongan dari web sumber dengan menggunakan curl. Proses selanjutnya adalah indeksi dimana proses ini akan membuat indek pencarian untuk lowongan hasil scraping. Proses ketiga adalah menyimpan lowongan dan indek lowongan ke dalam basis data. Proses terakhir adalah proses pencarian lowongan.

Scraping

Pada proses pengambilan (*scraping*) informasi dari sumber memiliki beberapa alur proses yang dapat dilihat pada gambar 2. Proses *scraping* diawali dengan proses otentifikasi data url. Curl *merequest* halaman lowongan web dan menyimpanya dalam bentuk *source code* html. Proses selanjutnya adalah *parsing* data. Proses ini akan mengambil data yang diperlukan dan selanjutnya disimpan ke variabel baru. Pengecekan dilakukan untuk memastikan tidak ada data lowongan yang sama. Proses selanjutnya adalah menghapus tanda baca dan kata hubung sehingga diperoleh hasil perkata.

Gambar 2. Alur Scraping

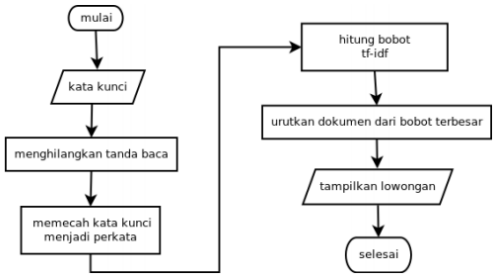


Pada proses *Scraping* akan dilakukan proses tokenisasi dan *stopword*. Tokenisasi adalah proses untuk membagi atau memecah teks yang dapat berupa kalimat, paragraf atau dokumen, menjadi token atau bagian tertentu seperti kumpulan kata dengan cara menghilangkan tanda baca atau mengubah huruf capital menjadi huruf kecil, sementara s*topwords* umumnya dimanfaatkan dalam *task information retrieval*.. Contoh *stopwords* untuk bahasa Inggris diantaranya "of", "the". Sedangkan untuk bahasa Indonesia diantaranya "yang", "di", "ke" [10].

Proses Pencarian

Proses pencarian dengan menggunakan metode pembobotan kata memiliki beberapa langkah yang dapat dilihat pada gambar 3. Pembobotan dimulai dengan menghilangkan tanda baca dan kata hubung. Selanjutnya proses yang dilakukan adalah mengubah kalimat menjadi perkata. Kata yang dihasilkan kemudian dicari ke dalam tabel indeks dan berikutnya menghitung nilai idf dan bobot tf-idf.

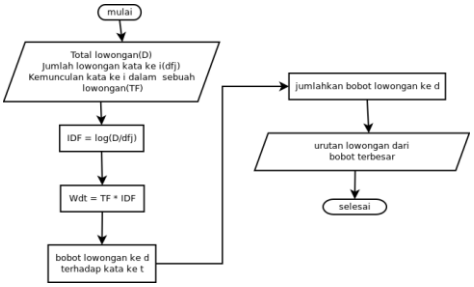
Gambar 3. Alur proses pencarian



Pembobotan Kata dengan TF-IDF

Term frequency (TF) merupakan metode yang paling sederhana dalam membobotkan setiap *term*. Setiap *term* diasumsikan memiliki kepentingan yang proporsional terhadap jumlah kemunculan *term* pada dokumen. Bobot kata (*term*) pada lowongan adalah frekuensi kemunculan *term* ke-i pada lowongan ke-j. Berbeda dengan TF yang memperhatkian tingkat kemunculan kata pada sebuah lowongan, IDF akan memperkatikan tingkat kemunculan *term* pada kumpulan lowongan. Semakin sedikit lowongan yang memiliki kemunculan *term* maka nilai IDF akan semakin besar. Untuk gambaran proses pembobotan dapat dilihat pada gambar 4.

Gambar 4. Alur proses pembobotan kata dengan TF-IDF



HASIL DAN PEMBAHASAN

Pada sistem yang dibangun, pencarian lowongan dapat dilakukan berdasarkan kota maupun pekerjaan. Gambar 5 menunjukkan hasil pencarian dari sistem.

Gambar 5. Hasil Pencarian

a. Hasil pencarian tanpa kata kunci b. Hasil pencarian dengan kata kunci

Gambar 5 (b) merupakan hasil pencarian lowongan dengan kata kunci "marketing guru admin". Sistem akan memulai dengan mencari lowongan yang memiliki kata yang sama dengan kata kunci yang digunakan. Seperti pada gambar 5 (b), sistem akan menampilkan 3 lowongan dimana masing-masing judul lowongan memiliki kata yang sama pada kata kunci, yaitu "marketing", "guru", dan "admin". Lowongan yang memiliki kemunculan kata paling banyak akan ditampilan terlebih dahulu, diikuti dengan lowongan yang memiliki kemunculan kata yang lebih sedikit.

Penghitungan bobot untuk setiap kata kunci dapat dijelaskan sebagai berikut. Misalkan pada basis data terdapat judul lowongan seperti yang ada pada tabel 1 dengan kata kunci pencarian "admin hrd Yogyakarta".

Tabel 1. Data lowongan

No	Judul Lowongan
1	Guru bahasa Inggris Yogyakarta
2	Admin hrd
3	Admin finance & accounting
4	Manager - Gamezone Malang

Sistem akan mencari lowongan yang memiliki kesamaan dengan kata kunci pencarian seperti yang ada pada Tabel 2.

Tabel 2. Kesamaan kata pencarian dan lowongan

Kata kunci	Kemunculan kata	Keterangan
Admin	2	Lowongan 2 dan lowongan 3
Hrd	1	Lowongan 2
Yogyakarta	1	Lowongan 1

Jumlah kemunculan kata yang ada pada tabel 2 akan menjadi nilai df untuk tiap kata pada kata kunci. Selanjutnya akan dihitung nilai idf dan bobot lowongan terhadap masing-masing kata kunci ke-i.

Untuk kata kunci "Admin"

$IDF_j = \log(N/df_j)$

$IDF_1 = \log(N/df_1)$

$= \log(4/2)$

$= 0301.$

Kata kunci ke-1 (T1) hanya muncul pada lowongan 2 dan 3 maka perhitungannya adalah sebagai berikut.

$W_{dt} = TF_{dt} * IDF_t$

$W_{12} = TF_{12} * IDF_1$

$= 1 * 0{,}301$

$= 0{,}301$

$W_{13} = TF_{13} * IDF_1$

$= 1 * 0{,}301$

$= 0{,}301$

Untuk kata kunci "Hrd"

$IDF_j = \log(N/df_j)$

$IDF_2 = \log(4/1)$

$= 0{,}602.$

Kata kunci ke-2 (T2) hanya muncul pada lowongan 2 maka perhitungannya sebagai berikut

$W_{22} = TF_{22} * IDF_2$

$= 1 * 0{,}602$

$= 0{,}602$

Untuk kata kunci "Yogyakarta"

$IDF_j = \log(N/df_j)$

$IDF_3 = \log(4/1)$

$= 0{,}602$

Kata kunci ke-3 (T3) hanya muncul pada lowongan 1 maka perhitungannya sebagai berikut

$W_{31} = TF_{31} * IDF_3$

$= 1 * 0{,}602$

$= 0{,}602$

ISSN 2798-3471 (Cetak)
ISSN 2798-3641 (Online)

Dengan demikian dapat diperoleh nilai bobot(w) untuk setiap kata kunci pada masing-masing lowongan seperti yang ada pada Tabel 3.

Tabel 3. Bobot kata pada dokumen

Term(kata)	Kemunculan kata				df	d/df	IDF	Bobot kata pada dokumen			
	L1	L2	L3	L4				L1	L2	L3	L4
Admin	0	1	1	0	2	4/2	0,301	0	0,301	0,301	0
Hrd	0	1	0	0	1	4/1	0,602	0	0,602	0	0
Yogyakarta	1	0	0	0	1	4/1	0,602	0,602	0	0	0
Jumlah bobot tiap lowongan								0,602	0,903	0,301	0

PENUTUP

Kesimpulan

Dari hasil dan pembahasan yang telah diuraikan dapat diperoleh beberapa kesimpulan, yaitu:

1. Penggunaan bobot TF-IDF dapat meranking lowongan yang dianggap lebih sesuai dengan lowongan yang dicari, melalui pembobotan kata-kata yang sesuai antara lowongan dengan kata kunci pencarian.
2. Pada proses *parsing* perlu dideskripsikan secara rinci tag html setiap web sumber agar data yang diambil seuai dengan kepentingan pencarian lowongan.

DAFTAR PUSTAKA

[1] Deolika, A., Kusrini, and Luthfi, E.T., 2019, Analisis Pembobotan Kata pada Klasifikasi Text Mining, *Jurnal Teknologi Informasi*, Vol. 3, No. 2, hal 179-184.

[2] Herwijayanti, B., Ratnawati, D. E., and Muflikhah, L., 2018, Klasifikasi Berita Online dengan menggunakan Pembobotan TF-IDF dan Cosine Similarity, *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vol. 2, No. 1, hal 306-312.

[3] Melita, R., Amrizal, V., Suseno, H. B., Dirjam, T., 2018, Penerapan Metode Term Frequency Inverse Document Frequency (Tf-Idf) dan Cosine Similarity pada Sistem Temu Kembali Informasi untuk Mengetahui Syarah Hadits Berbasis Web (Studi Kasus: Syarah Umdatil Ahkam), *Jurnal Teknik Informatika*, Vol. 11, No. 2, hal 149-165.

[4] Saadah, M. N., Atmagi, R. W., Rahayu, D. S., and Arifin, A.Z., 2013, Sistem Temu Kembali Dokumen Teks dengan Pembobotan Tf-Idf Dan LCS, *Jurnal Ilmiah Teknologi Informasi,* Vol. 11, No. 1, hal 17-20

[5] Liu, Q., Zhang, D., Wang, J., Yang, Y., Wang, N., 2018, Text Features Extraction based on TF-IDF Associating Semantic, IEEE 4th International Conference on Computer and Communications, 2338-2343

[6] Wu, H. C., Luk, R. W. P., Wong, K. F., and Kwok, K.L., Interpreting TF-IDF Term Weights as Making Relevance Decisions, *ACM Transactions on Information Systems*, Vol. 26, No. 3, Article 13, Publication date: June 2008,13:1 - 13:37

[7] Saadah M.N., Atmagi, R.W., Rahayu, D. S., Arifin, A.Z., 2013, Information Retrieval of Text Document with Weighting Tf-Idf and Lcs, *Journal of Computer Science and Information*, Volume 6, Issue 1, hal 34-37.

[8] Turland, Matthew, 2010, *Php|architect's Guide to Web Scraping*, Canada: Marco Tabini & Associates.

[9] Kurniawati, D., Triawan, D., 2017, Increased Information Retrieval Capabilities on e-commerce Websites using Scraping Techniques, International Conference on Sustainable Information Engineering and Technology (SIET), Malang, Nov 24-25.

[10] Rofiqi, M. A., Fauzan, A. C., Agustin, A. P., Ahmad Agung Saputra, A. A., Fahma, H. D., 2019, Implementasi *Term-Frequency Inverse Document Frequency (TF-IDF)* untuk Mencari Relevansi Dokumen Berdasarkan *Query*, Journal of Computer Science and Applied Informatics, Vol. 1, No. 2, hal 58-64

HALAMAN INI SENGAJA DIKOSONGKAN

ISSN 2798-3471 (Cetak)
ISSN 2798-3641 (Online)