SISTEM REKOMENDASI INDEKS *WEB* DENGAN METODE *FREQUENT TERMS* BERBASIS *MULTI INSTANCE LEARNING*

Darlis Herumurti, Joko Lianto Buliali, Ria Andriana
Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya
Email: darlis@its-sby.edu, joko@its-sby.edu, darko_chan@inf.its-sby.edu

ABSTRAK: Halaman indeks dikenal sebagai halaman yang mengelompokkan informasi-informasi, dengan memberikan judul serta penjelasan singkat tentang suatu informasi, dimana informasi lengkap akan dipresentasikan pada halaman-halaman lain. Namun dengan ketersediaan informasi yang menjadi semakin menumpuk, keberadaan halaman indeks yang semakin banyak justru menyebabkan kesulitan dalam mendapatkan informasi karena mungkin akan mengarahkan pada banyak informasi yang tidak relevan. Tanpa adanya sebuah sistem yang dapat membantu navigasi *user*, untuk mencari informasi yang diinginkan sama saja dengan sebuah kegiatan *trial* dan *error*. Dalam penelitian ini, dirancang sebuah sistem rekomendasi indeks web yang melibatkan aktifitas *user* dalam mengakses halaman indeks. Sistem ini mengelompokkan *frequent terms* pada halaman indeks dan kemudian mengimplementasikan metode Multi *Instance Learning* untuk memberikan rekomendasi secara otomatis dari halaman-halaman indeks baru. Algoritma yang digunakan adalah algoritma Citation kNN yang diadaptasi menjadi fretCit-kNN dengan mengaplikasikan minimal Hausdorff distance dalam pengukuran jaraknya. Dalam hasil proses dan analisis disimpulkan bahwa dengan beberapa macam uji coba data dari beberapa *user* sistem menampilkan performa hingga rata-rata 82,41% akurasi dan nilai kembalian sebesar 66,71%.

Kata kunci: halaman indeks, sistem rekomendasi, *multi instance learning, citation kNN, hausdorff distance.*

ABSTRACT: *Web index page is well known as page that arranges information by giving the title and short explanation about the information, where the complete information will be presented in other page. However since the amount of information become accumulate, the existence of a lot of index page exactly cause difficulty on getting information because it is possible to direct users into a mount of irrelevant information. Without a system which can help user navigation, the process of seeking the expected information is equal to a trial and error processing. In this paper, web index recommendation system is investigated which involved the activity of user on accessing the index page. This system will arrange the frequent term in index page and then implement Multi Instance Learning to give recommendation of the new index page automatically. The algorithm is citation kNN that will be adapted into fretCit kNN by implementing the minimal Hausdorff distance in measuring the distance. The experiments show that from the several test of users, the system give performance in average recommendation until 82,41% accuracy with 66,71% recall.*

Keywords: *index page, recommendation system, multi instance learning, citation kNN, hausdorff distance.*

PENDAHULUAN

Ada bermacam jenis halaman *web* pada Internet. Diantaranya ada yang hanya berisi indeks-indeks yang terdiri atas judul beserta penjelasan singkat tentang suatu informasi, sedangkan informasi lengkap dipresentasikan pada halaman lain yang terhubung dari indeks tersebut. Halaman *web* yang seperti ini dinamakan halaman indeks.

Seorang *user* dapat mengakses banyak halaman indeks, beberapa indeks diantaranya memiliki informasi yang menarik perhatian *user*, sementara yang lainnya tidak. Informasi sebenarnya bisa didapatkan dengan *browsing* dan *keyword searching*. Namun dengan adanya pertambahan jumlah situs dan halaman-halamannya, ketersediaan informasi menjadi semakin menumpuk, sehingga menyebabkan dua cara tersebut memiliki keterbatasan. *Browsing* menyebabkan kesulitan dalam mendapatkan informasi karena justru akan mengarahkan pada banyak *link* yang terdapat pada halaman-halaman *web*, sedangkan *keyword searching* akan menghasilkan begitu banyak informasi yang tidak relevan. Tanpa adanya sebuah sistem yang dapat membantu navigasi *user*, untuk mencari informasi yang diinginkan sama saja dengan sebuah kegiatan *trial* dan *error*.

Hal ini akan menjadi efektif apabila kemudian halaman-halaman indeks tersebut dapat secara otomatis dianalisis dalam sebuah sistem, sehingga hanya indeks-indeks yang berisi informasi yang relevan bagi *user* yang akan direkomendasikan. Karena setiap individu dapat memiliki ketertarikan yang berbeda-

beda, maka rekomendasi pun seharusnya dapat dibuat sesuai dengan analisis pada tiap individu.

Penelitian ini memfokuskan pada target analisis terhadap aktifitas *user* secara individual pada halaman indeks yang diakses, mengelompokkan *frequent terms* pada halaman tersebut dan kemudian mengimplementasikan metode *Multi Instance Learning* untuk memberikan rekomendasi secara otomatis dari halaman-halaman indeks baru.

Tujuan dari penelitian ini adalah untuk merancang sebuah sistem rekomendasi indeks untuk halaman *web* berdasarkan analisis aktifitas *user* secara individual dan *frequent terms* pada halaman yang diakses. Diharapkan, dengan merancang sebuah rekomendasi, akan membantu agar navigasi *user* menjadi lebih efektif dan efisien, serta informasi yang lebih relevan dan dibutuhkan *user* dapat dengan lebih mudah didapat.

Dalam pengerjaan penelitian ini, permasalahan yang akan dibahas adalah:

1. Bagaimana mendapatkan *Frequent terms* pada halaman-halaman indeks.
2. Bagaimana mengimplementasikan sebuah sistem rekomendasi indeks berbasis metode *Multi Instance Learning*.
3. Bagaimana menentukan rekomendasi halaman indeks yang sesuai dengan minat *user*.

MULTI *INSTANCE LEARNING*

Multiple instance problem, ataupun *multi instance learning* adalah sebuah *framework* yang unik, dikarenakan dalam *multi instance learning*, yang diberi label adalah cukup sekelompok sampel/*instance* (yang disebut juga *bag*), dan bukanlah memberikan label pada setiap *instance* seperti pada *supervised learning*. Karena keunikannya inilah, metode *multi instance* telah mendapatkan begitu banyak perhatian dalam komunitas pengembangan *Machine Learning* dan kemudian dinyatakan sebagai *learning framework* yang baru [2].

Multi instance learning pertama kali diperkenalkan oleh Dietterich et al. (1997) dalam penelitian terhadap prediksi aktivitas obat. Dalam *framework* ini, *training set* terdiri dari banyak *bag* dimana setiap *bag* merupakan kumpulan dari banyak *instance*. Lalu, label akan diberikan pada setiap *bag*, bukan terhadap setiap *instance*. Karena itu, algoritma pembelajaran untuk metode ini harus membuat sebuah *classifier* yang akan mengklasifikasikan sampel-sampel (yaitu *bag* dari *instance*) yang belum diketahui labelnya, secara tepat.

Tujuan dari penelitian Dietterich adalah untuk membuat sistem pembelajaran dengan kemampuan untuk memprediksikan apakah suatu molekul baik untuk digunakan dalam pembuatan obat, dengan menganalisis sekumpulan molekul yang diketahui kondisinya. Molekul yang kondisinya baik, adalah yang salah satu dari *low-energy shapes*-nya memiliki keterikatan dengan target. Sebelumnya pada *supervised learning*, seluruh molekul yang baik akan diberikan label positif, dan sebaliknya negatif. Ternyata, Dietterich menunjukkan bahwa metode ini masih mengalami banyak *false positive noise* atau nilai positif yang salah, karena mungkin saja sebuah molekul yang baik memiliki banyak *low energy shapes*, namun hanya satu yang benar-benar dalam kondisi yang bagus. Hal ini akan menyebabkan molekul tersebut dianggap tidak baik. Untuk menangani masalah ini, Diettrich menyatakan molekul sebagai *bag* dan *low energy shape* sebagai *instance* dari *bag*, lalu memformulasikan *multi instance learning*.

Banyak jenis dari algoritma untuk *multi instance learning*, diantaranya *Diverse Density* yang diusulkan oleh Maron dan Lozano-Pérez, k-*Nearest Neighbor* yang diperluas oleh Wang dan Zucker, *multi instance decision tree Relic* oleh Ruffo, *multi-instance decision tree ID3-MI* dan *rule inducer RIPPER-MI* milik Chevaleyre dan Zucker, *multi-instance neural network BP-MIP* oleh Zhou dan Zhang, juga *EM-DD* dari Zhang dan Goldman.

PENDEKATAN *LAZY LEARNING* PADA MASALAH MULTI *INSTANCE*

Banyak dari masalah *multi instance* justru diimplementasikan dalam *eager learning*, dibandingkan dari *lazy learning*. Di tahun 2000, Wang dan Zucker memberikan pandangan lain dan membuktikan bahwa algoritma *lazy learning* dapat diadaptasi ke dalam masalah *multi instance* dengan mengaplikasikan *hausdorff distance* pada algoritma kNN. Hasil yang dicapai, justru memberikan hasil terbaik pada masalah pembuatan obat, dengan akurasi 92,4 % untuk algoritma Citation-kNN. Ini telah membuktikan bahwa *multi instance* dapat pula diimplementasikan dengan pendekatan *lazy learning*.

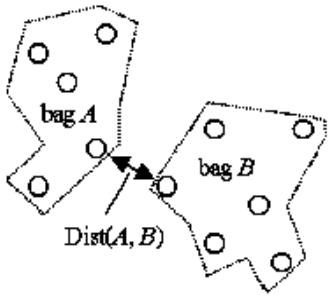
K-*NEAREST NEIGHBOR* DAN MULTI *INSTANCE LEARNING*

Untuk setiap algoritma *nearest neighbor*, kuncinya adalah dengan mendefinisikan metrik untuk menghitung jarak dari objek yang berbeda. Namun, untuk mengadaptasi kNN terhadap masalah *multi instance*, fungsi kNN yang standar harus dimodifikasi, dari yang mendiskriminasikan *instance*, menjadi diskriminasi terhadap *bag* [3]. Dalam diskriminasi

instance, untuk jarak antara 2 vektor a dan b, cukup digambarkan dengan:

$$Dist(A,B)=\|\mathbf{a}-\mathbf{b}\|$$

yang merupakan persamaan *Euclidean distance*. Namun, karena tujuan dari algoritma ini haruslah untuk mendiskriminasikan *bag*, maka persamaan tersebut harus lebih diperluas agar dapat mengukur jarak antara dua *bag*. Ilustrasi dalam menentukan jarak antara dua *bag* ditunjukkan pada Gambar 1.



Gambar 1. Ilustrasi jarak antara dua bag

Misalkan terdapat dua *bag* A = {a1,a2,...,am} dan B = {b1,b2,...,bn} dimana *ai* ($1 \le i \le m$) dan *bj* ($1 \le j \le n$) adalah *instance*-nya. Pada gambar 1, diilustrasikan bahwa terdapat dua set vektor, yang masing-masing A dan B adalah sebuah vektor dalam *space*. Jarak A dan B adalah jarak yang terdekat dari dua instance-nya. Atau dapat dituliskan dalam persamaan :

$$Dist(A,B)=\min_{\substack{1\le i\le m\\ 1\le j\le n}}(Dist(ai,bj)) = \min_{a\in A}\min_{b\in B}\|a-b\| \quad (1)$$

yang ternyata adalah persamaan yang sesuai dengan minimum *Hausdorff distance*.

Karena itu dapat disimpulkan, bahwa dengan menerapkan minimal *Hausdorff distance* untuk jarak antara *bag*, maka metode lama dari *k*NN dapat diimplementasikan dalam masalah *multi instance*. Namun, Wang dan Zucker memberikan bukti bahwa hal itu saja belum cukup dengan eksperimen pada data dalam aplikasi pembuatan obat.

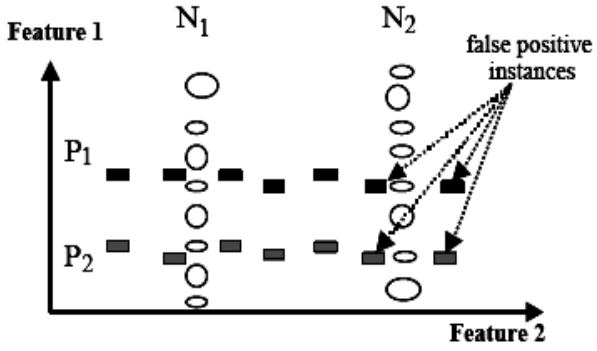
Dalam aplikasi ini, terdapat 47 *bag* yang positif dan 45 negatif. Dengan metode *k*NN, *class* dari *bag* yang belum memiliki label, adalah *class* yang paling banyak dimiliki oleh *training bag* terdekatnya. Misalkan dengan (K=3) *nearest neighbor*, *class*-nya adalah {*P*,*P*,*N*} (*P* adalah positif dan *N* adalah negatif), maka *class* dari *bag* tersebut diprediksikan sebagai *P*. Namun strategi ini ternyata tidak optimal, karena masih terdapat distribusi *class* yang tidak tepat.

Tabel 1 menunjukkan seluruh *class* dengan metode k-*nearest neighbor* dan *Hausdorff distance*. Kolom ketiga dan keempat menunjukkan jumlah *bag* yang memiliki set *class* pada kolom kedua. Terlihat ada kontradiksi ketika k=3, yaitu 18 *bag* yang memiliki kelas {P,P,N} untuk *nearest neighbor*-nya. Dari 18 *bag* ini, 13 diantaranya bernilai negatif, dan 5 bernilai positif. Padahal seharusnya, jika *bag* ini dinyatakan sebagai positif daripada negatif, maka nilai akurasi dari prediksi ini akan menjadi lebih tinggi.

Tabel 1. Distribusi class dari k Nearest Neighbor untuk Bag Positif/Negatif dari Set Data Aplikasi Pembuatan Obat

K	K nearest neighbors	Number of positive	Number of negative	Sum
1	{*P*}	41	9	50
	{*N*}	6	36	42
2	{*P*,*P*}	41	3	44
	{*P*,*N*}	5	15	20
	{*N*,*N*}	1	27	28
3	{*P*,*P*,*P*}	40	2	42
	{*P*,*P*,*N*}	**5**	**13**	**18**
	{*P*,*N*,*N*}	2	9	11
	{*N*,*N*,*N*}	0	21	21

Dalam metode *supervised learning*, kontradiksi ini tidak akan terjadi. Penjelasan untuk mengapa terjadi kontradiksi tersebut adalah karena kemungkinan terdapat '*false positive instance*' atau *instance* yang bernilai positif, namun ternyata berada di sekitar *instance* dari *bag-bag* yang bernilai negatif, sehingga menimbulkan kesalahan prediksi terhadap *bag*.



Gambar 2. Kumpulan Instance, Instance dari Bag Negatif digambarkan dengan Lingkaran, Bag Positif dengan Persegi

Dari Gambar 2, mengilustrasikan adanya *false positive instance* yang memberikan penjelasan terhadap kontradiksi yang terjadi. Jika {P1,P2,N1} diberikan sebagai 3 *training bag* untuk N2, dan N2 akan diprediksikan sebagai positif. Sebaliknya, jika diberikan {N1, N2, P1} sebagai 3 *training bag*, P2 akan diprediksikan sebagai negatif.

Karena itu, Wang dan Zucker memberikan beberapa cara untuk mengatasi masalah klasifikasi ini. Salah satu caranya adalah dengan memberikan bobot lebih kepada *bag* yang berlabel negatif. Namun pendekatan kedua bisa menjadi metode adaptasi yang lebih baik untuk kNN dalam masalah multi instance, yang dinamakan Citation - kNN.

CITATION KNN

Algoritma Citation–*k*NN adalah algoritma *nearest neighbor*, yang memberikan label pada *bag* dengan menganalisis bukan hanya *bag* terdekat dari *bag* tersebut, tapi juga dengan melihat *bag* lain yang menganggapnya sebagai *bag* terdekat [1]. Biasanya, di dalam pengelompokan artikel, kita bisa pula melihat dokumen-dokumen yang berelasi dengan artikel tersebut, atau disebut dengan *reference*. Selain itu, ada juga yang dinamakan *citer*, yaitu dokumen-dokumen lain yang menganggap artikel itu sebagai referensinya. Itu berarti artikel tersebut, selain berelasi dengan *reference*-nya, juga memiliki relasi dengan *citer*-nya. Ini menjadi konsep utama dari algoritma Citation – kNN.

Misalkan, diberikan 6 *bag* yaitu {b1,b2,b3,b4, b5,b6} pada *set data*. Untuk *nearest neighbor* dari tiap *bag* ditunjukkan pada Tabel 2.

Tabel 2. Nearest neighbor dari 6 Bag {b1,b2,b3, b4,b5,b6}

	N =1	N =2	N =3	N =4	N =5
b1	*B3*	*b2*	B5	b4	b6
b2	**B1**	b4	B5	b3	b6
b3	B5	**b1**	B2	b6	b4
b4	B6	b2	B1	B3	b5
b5	**B1**	b2	B3	B6	b4
b6	B4	b3	B1	B2	b5

Pada tabel 2, N adalah tingkatan terdekat dari *bag*. Jika *reference* yang terdekat (R-*reference*) dan *citer* yang terdekat (C-*citer*) dari *bag* diberikan nilai 2, maka untuk *bag* b1, R-*reference*-nya adalah {b3,b2} dan C-*citer*-nya adalah {b2,b3,b5}. Lalu, kedua objek ini haruslah dikombinasikan untuk memprediksikan *class*-nya. Diasumsikan jumlah *bag* yang positif dari R adalah Rp dan untuk yang negatif adalah Rn. Begitu juga untuk C, jumlah *bag* yang berlabel positif adalah Cp dan negatif adalah Cn.

Untuk setiap algoritma *nearest neighbor*, kuncinya adalah dengan mendefinisikan metrik untuk menghitung jarak dari objek yang berbeda. Untuk Citation-kNN, *minimal Hausdorff distance* dapat diaplikasikan untuk memodifikasi fungsi kNN yang standar, dari yang mendiskriminasikan *instance*, menjadi diskriminasi terhadap *bag*.

Tabel 3. Distribusi dari Bag Positif dan Negatif pada R dan C dari Bag yang belum memiliki Label

	***Bag* Positif**	***Bag* Negatif**	
Reference	Rp	Rn	R
Citer	Cp	Cn	N
	P = Rp+Cp	N = Rn+Cn	

Karena itu Rp, Rn, Cp, Cn dikomputasi dengan *Hausdorff distance* dan klasifikasinya didefinisikan sebagai berikut:

- Jika p > n, maka *class* dari bag b diprediksikan sebagai positif.
- Jika p < n, maka *class* dari bag b diprediksikan sebagai negatif.
- Jika p = n, maka *class* dari bag b diprediksikan sebagai negatif.

Alasan mengapa *class* dari *bag* b diprediksikan negatif jika p = n, adalah karena ada kemungkinan dari munculnya *false positive instance*. Untuk kapabilitas dari Citation-kNN, Wang dan Zucker memberikan eksperimen pada set data aplikasi pembuatan obat dan algoritma ini memiliki akurasi sebesar 92,4 %.

SISTEM REKOMENDASI INDEKS WEB

Dari berbagai macam halaman *web* dalam dunia internet, ada diantaranya halaman-halaman yang mengelompokkan informasi dengan cara menyediakan judul serta ringkasan saja, sedangkan detil informasi akan diberikan di halaman lain yang dihubungkan dengan *hyperlink*. Halaman ini dinamakan halaman indeks web.

Ketika penggunaan internet semakin ramai dan penyediaan informasi yang semakin beragam, jumlah dari halaman-halaman indeks pun bertambah, sehingga tujuannya untuk mengurangi beban *user* dalam pencarian informasi yang spesifik kembali menjadi abstrak. Akhirnya, *user* tetap harus melakukan pencarian terhadap jenis halaman indeks yang di inginkan. Maka, Zhi-Hua Zhou, Kai Jiang dan Ming Li memberikan suatu solusi analitik terhadap masalah ini dengan pandangan *multi instance* [3].

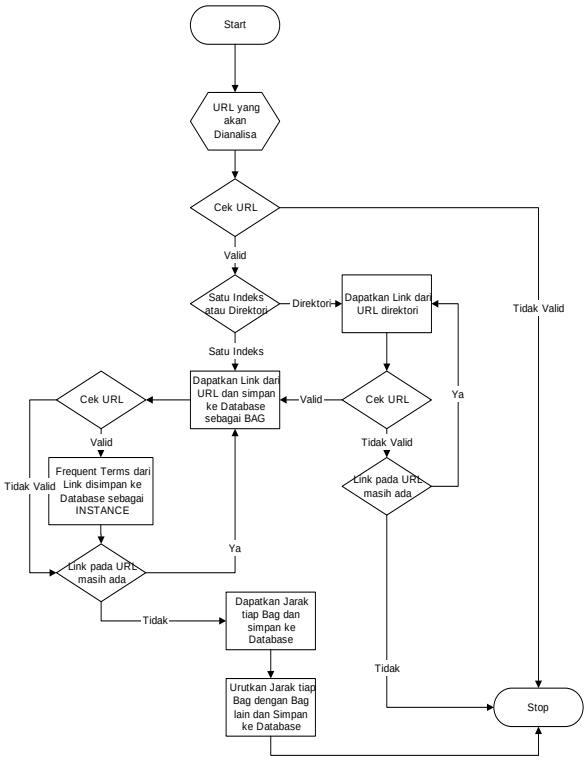
Solusi ini adalah dengan mempelajari halaman indeks yang telah dikunjungi seorang individu dan kemudian mengidentifikasi apakah rekomendasi halaman indeks baru akan menarik perhatiannya. Pada dasarnya, tujuannya adalah untuk memberikan label pada halaman indeks yang belum dikunjungi individu tersebut, apakah berlabel positif atau negatif. Halaman indeks yang positif adalah halaman yang setidaknya memiliki satu *link* informasi yang diminati individu. Sebaliknya, jika tidak satu pun dari informasi yang

ditawarkan halaman indeks tidak diminati oleh individu, maka diberikan label negatif. Setiap halaman indeks dinyatakan sebagai sebuah *bag*, dimana *instance*-nya adalah seluruh halaman yang terhubung pada halaman indeks tersebut.

PERANCANGAN SISTEM

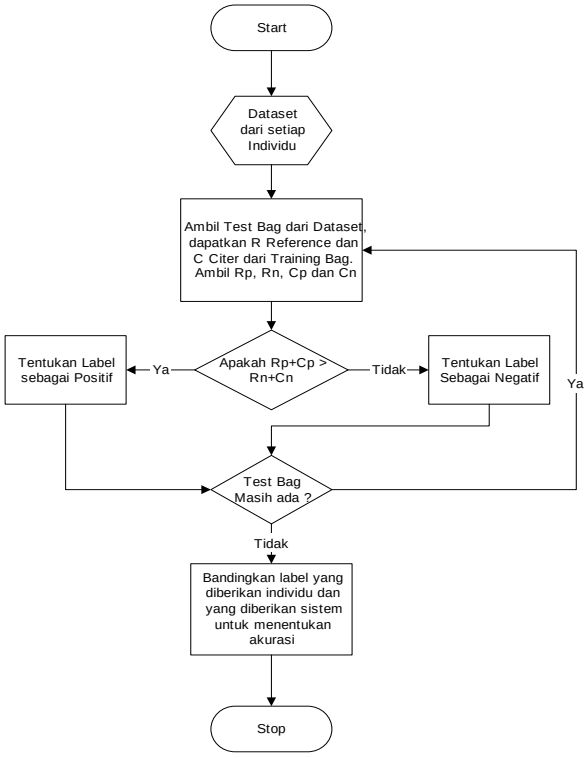
Sistem yang akan diimplementasikan pada Penelitian ini adalah sebuah sistem yang berupa sebuah *intelligent system* untuk memberikan rekomendasi kepada user. Analisis akan diberikan kepada tiap *user* secara individual, untuk mengetahui apakah sebuah halaman indeks baru akan menarik perhatiannya. Label positif diberikan pada halaman indeks yang setidaknya satu atau lebih *link*-nya menarik perhatian *user*, sebaliknya akan diberikan label negatif. Data yang diambil dari tiap *instance* adalah 20 *frequent terms* atau terminologi yang kerap muncul pada URL tersebut. Setelah itu, diterapkan perhitungan jarak, sehingga label R-*reference* dan C-*citer* terdekat dapat diambil, dan digunakan untuk penentuan label bag yang dianalisis.

Alur kerja dari sistem dapat dipecah menjadi dua proses, yaitu proses pengambilan data dan proses analisis. *Flowchart* dari aplikasi untuk proses pengambilan data ditunjukkan pada Gambar 3:



Gambar 3. Flowchart Aplikasi Proses Pengambilan Data

Sedangkan *flowchart* untuk proses analisis ditunjukkan pada Gambar 4.



Gambar 4. Flowchart Aplikasi: Proses Analisis

Dalam pengambilan *frequent term* dari suatu URL, yang pertama kali dilakukan adalah dengan mendapatan *source* HTML dari URL tersebut. Karena yang diperlukan adalah informasi dari URL tersebut dan bukan *script* dari *web* tersebut, maka ada beberapa hal yang patut diamati dalam pengambilan *source* HTML ini:

1. Harus dilakukan scan terhadap *tag-tag* HTML, *script* dari JavaScript dan CSS. Ini dapat dilakukan dengan melakukan *regular expression*.
   a. *Tag* HTML diawali dengan <*variabel* dan diakhiri dengan >, maka *regular expression*-nya dapat dinyatakan seperti : <[^>]+>
   b. JavaScript diawali dengan <script *script Javascript* dan diakhiri dengan </script>, maka *regular expression*-nya dapat dinyatakan seperti : <script.* /script>
   c. *Style* pada CSS diawali dengan <style *CSS* dan diakhiri dengan </style>, maka *regular expression*-nya dapat dinyatakan seperti: <style.*/style>
   d. Untuk bagian *head* dari HTML tidak akan digunakan, karena informasi yang diperlukan terdapat pada *body* HTML tersebut. Untuk mengatasi hal ini, dapat diberikan *regular expression* : <head.*/head>

2. Dalam *source* yang telah di-*scan*, ada kemungkinan terdapat informasi yang tidak perlu berupa kata-kata dengan frekuensi yang tinggi, namun bukan kata-kata inti dari halaman tersebut, seperti kata-kata penghubung, atau pelengkap pada sebuah kalimat. Contohnya : *a, about, also, am, an, and, are, as, at, b, be, been, but, by, can, com, could, didn't, do, doesn't, don't, during, for, from, had, has, have, he, her, here, him, his, i, if, in, is, it, just, m, me, might, no, not, of, on, or, our, out, over, she, so, still, td, that, the, their, them, there, they, this, to, too, us, was, we, were, what, where, when, who, whose, will, with, would, you, your*. Maka seluruh kata-kata ini harus diabaikan agar tidak terjadi informasi yang rancu.
3. Abaikan pula karakter-karakter yang tidak perlu seperti : *+-/><;:',",%20,* dll .

Setelah *source* HTML tersebut menjadi sebuah kumpulan informasi yang lebih relevan, untuk mendapatkan *frequent term* adalah dengan menghitung kemunculan tiap kata dan jumlahnya. Algoritmanya adalah sebagai berikut:

1. Masukkan setiap kata dalam *token*.
2. Inisialisasi i = 0
3. Inisialisasi sum_token = 0.
4. Bandingkan *token* i dengan *token* lain (misal *token* j, j dimulai dari i + 1), jika *token* i = *token* j dan i<j (bukanlah kata sebelumnya yang kemungkinan sudah dihitung), maka sum_token ditambah nilainya dengan 1 atau sum_token = sum_token + 1.
5. Maka kata pada *token* i memiliki jumlah sebanyak sum_token, data ini dimasukkan ke *array* baru.
6. Kembali ke inisialisasi pada nomor 3, dan terus diulang hingga jumlah i mencapai panjang maksimal *token* (semua kata telah dihitung).

FRETCIT KNN

Perhitungan jarak dalam algoritma Citation kNN adalah menggunakan minimal *Hausdorff distance*. Namun, selama *Hausdorff distance* ini masih mengaplikasikan *Euclidean distance* dalam persamaannya, maka perhitungan tersebut hanya bisa diaplikasikan dalam objek numerikal, atau *instance* dengan attribut yang numerik. Sedangkan untuk masalah rekomendasi indeks web, *instance* akan dideskripsikan dalam atribut yang tidak teratur, dan mungkin saja akan berupa tekstual. Untuk mengatasi hal ini, maka perhitungan jarak yang baru harus dikembangkan. Zhi Hua Zhou, Kai Jing dan Ming Li memberikan solusi dalam pengembangan dari minimal *Hausdorff Distance* ini [3].

Misalkan untuk dua set *frequent term* pada *bag* A dan B, yang masing-masing memiliki n *instance*, yaitu {a1,a2,...,an} dan {b1,b2,...,bn}. Secara intuitif, cara untuk mengukur jarak dari dua *bag* itu adalah dengan melihat berapa banyak terminologi yang sama antara kedua *bag* itu. Contohnya, jika A = {merah, putih, kuning}, B={hitam, merah, kuning}, dan C = {abu-abu, hijau, kuning}. Di sini sangat jelas jika A lebih dekat dengan B daripada dengan C, karena jumlah kata yang sama dari A dan B adalah 2, sedangkan A dan C hanya 1.

Dengan melihat kondisi tersebut, maka minimal *hausdorff distance* untuk set dari *frequent term* dapat dimodifikasi menjadi:

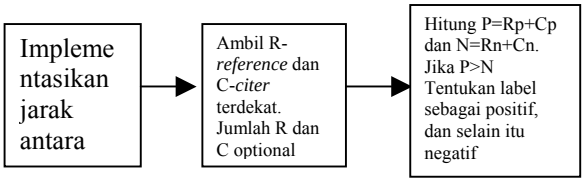
$$fret\text{-}\min_{a\in A, b\in B} H(A,B) = Min(1 - \sum_{\substack{i,j=1 \\ ai=bj}}^{n} 1/n) \qquad (2)$$

Dengan mengaplikasikan fret-minH(.) ini, jarak untuk *bag* A dan B dapat dikomputasi, yaitu fret-minH(A,B) = 1 – 2 (1/20) =0.9 dan untuk jarak *bag* A dan C adalah fret-minH(A,C) = 1 – 1 (1/20) = 0.95. Yang menandakan bahwa A dan B memiliki jarak yang lebih dekat daripada A dan C.

Maka, untuk perhitungan jarak dalam algoritma Fretcit-kNN, akan mengimplementasikan fret-min H(A,B) antara tiap *bag* dengan *bag* lain. Kemudian, R-*reference* dan C-*citer* dari *training bag* yang memiliki jarak terdekat dari *bag* tersebut akan digunakan dalam penentuan labelnya. Jika Rp+Cp > Rn + Cn maka label dinyatakan positif, dan selain itu akan dinyatakan sebagai negatif. Dengan algoritma ini, rekomendasi halaman indeks baru kepada individu dapat dilakukan. Namun, ada beberapa hal yang harus diperhatikan dalam perhitungan jarak antara tiap halaman indeks. Hal ini antara lain:

1. Jarak bernilai 0 antara *bag* yang diakibatkan ada dua *instance* dengan alamat URL yang sama, sehingga mengacu pada isi halaman yang sama. Hal ini harus diabaikan, karena diasumsikan seorang individu tidak akan tertarik untuk mengetahui informasi pada URL yang sama untuk kedua kalinya.
2. Jarak bernilai 0 karena pada dua *instance*, isi dari *frequent terms* serta jumlahnya sama. Ini dapat diartikan dalam URL yang berbeda, mungkin saja isi dari informasinya sama. Dengan asumsi yang sama, hal ini harus pula diabaikan.

Langkah-langkah Fretcit-kNN:



Gambar 5. Alur algoritma Fretcit-kNN

HASIL UJI COBA

Untuk set data yang digunakan dalam analisis, disiapkan 113 halaman indeks yang kemudian diberikan label oleh 7 individu sesuai dengan minat mereka masing-masing.

Masing-masing individu akan menentukan label untuk tiap halaman indeks. Label positif adalah apabila individu tersebut tertarik dengan salah satu bagian informasi atau lebih dalam halaman indeks tertentu. Jika individu tersebut tidak mengakses seluruh *link*, maka diartikan individu tersebut tidak tertarik dengan halaman indeks dan halaman indeks akan diberikan label negatif. Dalam 113 *bag* yang digunakan, secara acak akan dipilih 75 *bag* sebagai *training set* dan sisanya yaitu 38 *bag* akan digunakan sebagai *test set*. Jumlah *bag* positif dan negatif dari setiap individu ditabulasikan dalam Tabel 4.

Tabel 4. Jumlah Training dan Test Set Data dari Individu

	Training Set		Test Set	
	Positif	**Negatif**	**Positif**	**Negatif**
VOLUNTEER_1	17	58	4	34
VOLUNTEER_2	18	57	3	35
VOLUNTEER_3	14	61	7	31
VOLUNTEER_4	56	19	33	5
VOLUNTEER_5	62	13	27	11
VOLUNTEER_6	60	15	29	9
VOLUNTEER_7	39	36	16	22

Menggunakan set data yang telah disiapkan, akan diterapkan algoritma Citation kNN dengan memakai jumlah *reference* dan *citer* yang berbeda-beda untuk penentuan label setiap *bag*. Pada uji coba ini, diujikan 9 model analisis, yaitu 4 *Reference* dan 2 *Citer*, 4 *Reference* dan 4 *Citer*, 4 *Reference* dan 6 *Citer*, 6 *Reference* dan 2 *Citer*, 6 *Reference* dan 4 *Citer*, 6 *Reference* dan 6 *Citer*, 7 *Reference* dan 2 *Citer*, 7 *Reference* dan 4 *Citer*, serta 7 *Reference* dan 6 *Citer*.

Persentase dari akurasi seluruh individu (V1, V2, V3, V4, V5, V6, V7) untuk setiap *Reference-Citer* yang telah dianalisis, ditunjukkan dalam Tabel 5 (4-2 berarti 4 *Reference* 2 *Citer*).

Tabel 5. Persentase Akurasi Hasil Uji Coba

	4-2	**4-4**	**4-6**	**6-2**	**6-4**	**6-6**	**7-2**	**7-4**	**7-6**
V1	84%	79%	84%	89%	87%	89%	89%	87%	87%
V2	92%	92%	92%	87%	87%	87%	87%	87%	87%
V3	82%	84%	84%	82%	87%	82%	79%	82%	82%
V4	82%	82%	82%	84%	87%	84%	82%	84%	82%
V5	79%	79%	82%	74%	74%	74%	74%	74%	74%
V6	76%	79%	76%	79%	79%	79%	76%	79%	76%
V7	74%	71%	74%	76%	74%	71%	71%	71%	71%
Rata-rat	81,29%	80,86%	82%	81,57%	82,14%	**80,86%**	79,71%	80,57%	79,86%

Persentase dari nilai kembalian seluruh individu (V1, V2, V3, V4, V5, V6, V7) untuk setiap *Reference-Citer* yang telah dianalisis, ditunjukkan dalam Tabel 6 (4-2 berarti 4 *Reference* 2 *Citer*).

Tabel 6. Persentase Nilai Kembalian Hasil Uji Coba

	4-2	**4-4**	**4-6**	**6-2**	**6-4**	**6-6**	**7-2**	**7-4**	**7-6**
V1	33%	0%	25%	50%	33%	50%	50%	33%	33%
V2	50%	50%	50%	25%	25%	25%	25%	25%	25%
V3	50%	67%	67%	50%	100%	50%	33%	50%	50%
V4	88%	88%	88%	89%	89%	89%	88%	89%	88%
V5	77%	77%	79%	73%	73%	73%	73%	73%	73%
V6	79%	80%	79%	80%	80%	80%	79%	80%	79%
V7	67%	63%	67%	71%	67%	63%	65%	65%	63%
Rata-ra	63,43%	60,71%	65%	62,57%	**66,71%**	61,43%	59%	59,29%	58,71%

KESIMPULAN DAN SARAN

Penulisan tentang Sistem Rekomendasi Indeks Web ini memberikan aplikasi untuk teknik *web mining* menggunakan *multi instance learning*, yang juga memberikan solusi baru dalam melakukan *web mining*. Dalam sistem rekomendasi ini, jika dipandang dalam masalah *multi instance*, maka halaman indeks dapat dinyatakan sebagai *bag*, dimana seluruh *link* dan informasi di dalam *link* tersebut adalah *instance*-nya. Untuk mendapatkan *bag* yang sesuai dengan minat individu dan kemudian merekomendasikannya, digunakan algoritma kNN yang telah diadaptasi dalam *multi instance learning*, dengan pengukuran jarak yang mengimplementasikan minimal *Hausdorff distance* antara terminologi yang kerap muncul dalam *instance* (*frequent term)*. Algoritma kNN akan menggunakan baik *reference* (*bag* terdekat dari *bag* yang dianalisis) dan juga *citer* (*bag* yang menganggap *bag* yang dianalisis sebagai *bag* terdekat) dalam memberikan label kepada *bag* yang dianalisis. *Bag* yang memiliki label positif akan direkomendasikan kepada individu. Dengan metode ini, didapatkan hasil rata-rata akurasi tertinggi pada kondisi 6 *Reference* 4 *Citer* yaitu 82,14 % dan nilai kembalian sebesar 66,71%. Namun, ketika dilakukan penambahan ataupun pengurangan jumlah *Reference* dan *Citer*, performa sistem kembali menurun. Hal ini kemungkinan disebabkan oleh munculnya *noise* yang terjadi akibat terlalu banyak atau terlalu sedikit *bag* yang diperhitungkan.

Untuk pengembangannya, sistem rekomendasi indeks *web* dengan metode *frequent terms* berbasis *multi instance learning* ini dapat diimplementasikan pada aplikasi-aplikasi yang berhubungan langsung dengan *user* seperti misalnya *web browser* yang bersifat adaptif dengan aktifitas *user*. Selain itu, untuk metode pengambilan data halaman indeks sebaiknya diberikan fungsi *update* dalam setiap beberapa kurun

waktu, agar ketika terjadi perubahan informasi pada halaman indeks, informasi dalam sistem juga ikut berubah.

DAFTAR PUSTAKA

1. Wang, J., Zucker, J.D., *Solving the Multiple-Instance Problem: A Lazy Learning Approach*, Proceedings of the 17th International Conference of Machine Learning, San Francisco, CA, 2000.
2. Zhi Hua Zhou dan Min Ling Zhang. *Ensembles of Multi-Instance Learners*. Proceedings of the 14th European Conference on Machine Learning (ECML'03); Cavtat-Dubrovnik, Croatia, LNAI 2837: 2003.
3. Zhi Hua Zhou, Kai Jiang dan Ming Li., *Multi-Instance Learning Based Web Mining.* Applied Intelligence. 2005, 22(2): 135-147.