KLUSTERISASI DOKUMEN BERITA BERBAHASA INDONESIA MENGGUNAKAN DOCUMENT INDEX GRAPH

Sari Ernawati[1], Arie Ardiyanti, ST., MT.[1], Erwin Budi Setiawan[2]
[1]*Jurusan Teknik Informatika, Fakultas Teknik Informatika, IT Telkom Bandung*
[2]*Jurusan Ilmu Komunikasi, Fakultas Sains, IT Telkom Bandung*
E-mail: sari.ernawati@gmail.com ,rie006@yahoo.com, erw@ittelkom.ac.id

ABSTRAKS
Berita elektronik merupakan media informasi yang paling populer dan interaktif saat ini. Begitu interaktifnya, hingga perkembangannya cukup pesat. Terbukti bertambah banyaknya situs perusahaan maupun situs personal, yang berarti semakin meningkatkan jumlah informasi dan data. Peningkatan yang pesat ini juga dipacu oleh penggunaan internet yang semakin berkembang dibandingkan era sebelumnya. Sebagai akibatnya, jumlah informasi meningkat secara eksponensial. Banyaknya data yang ada, semestinya dapat memberikan manfaat yang banyak pula. Clustering merupakan salah satu metode untuk pengelompokan dokumen dengan menemukan keterkaitan antardokumen. Saat ini, kebanyakan metode klusterisasi hanya mengandalkan perhitungan kesamaan berdasarkan kata dan tidak memperhatikan aspek lain, misalnya kesamaan frasa, misalnya Vector Space Model. Pada makalah ini berusaha mengklusterkan dokumen dengan metode Document Index Graph yang menggunakan kombinasi dua kesamaan dokumen yaitu; kesamaan berbasis kata dan kesamaan berbasis frasa. Metode ini diuji coba dengan menggunakan sampel berita berbahasa Indonesia dari media massa berbasis web. Pemilihan fragmentation factor dan similarity threshold yang tepat akan meningkatkan kualitas kluster. Hasil klusterisasi dievaluasi berdasarkan nilai precision dan recall.

Kata Kunci: clustering, Document Index Graph, fragmentation factor, similarity threshold.

1. PENDAHULUAN

Berita elektronik merupakan media informasi yang paling populer dan interaktif saat ini. Begitu interaktifnya, hingga perkembangannya cukup pesat. Terbukti bertambah banyaknya situs perusahaan maupun situs personal, yang berarti semakin meningkatkan jumlah informasi dan data. Peningkatan yang pesat ini juga dipacu oleh penggunaan internet yang semakin berkembang dibandingkan era sebelumnya. Sebagai akibatnya, jumlah informasi meningkat secara exponensial, - lebih dari 550 triliun dokumen saat ini.

Banyaknya informasi semestinya dapat memberikan manfaat bagi *user*. Namun terkadang tidak mudah bagi *user* untuk mengakses setiap informasi yang berkaitan. Keterbatasan waktu maupun perangkat yang dimiliki merupakan salah satu penyebabnya. Oleh karena itu diperlukan sebuah metode untuk mengelompokkan dokumen tersebut agar memudahkan dalam pengambilan informasi sesuai kebutuhan *user*. Klusterisasi merupakan salah satu metode yang dapat digunakan untuk menemukan keterkaitan antar dokumen. Tujuan klusterisasi adalah untuk memisahkan sekumpulan dokumen ke dalam beberapa grup atau kluster dengan menilai kemiripan antar dokumen dari segi konten. Pengelompokan berita-berita yang saling berkait ini, akan membantu *user* untuk menemukan informasi yang dibutuhkan.

Banyak metode yang dapat dipakai dalam klusterisasi dokumen seperti dengan *Suffix Tree*, *Single Pass Clustering* maupun *K-Nearest Neigbour*. Kebanyakan metode klusterisasi dokumen berbasis *Vector Space Model* yang merepresentasikan dokumen sebagai fitur vektor dari term yang muncul pada semua dokumen. Setiap fitur vektor mengandung bobot term atau frekuensi term yang ada pada dokumen tersebut. Kesamaan antar dokumen dihitung menggunakan perhitungan yang berbasis fitur vektor, misalnya *cosine measure* dan *Jaccard measure.* Klusterisasi dengan metode seperti ini hanya memperhatikan analisis *single term*, tanpa memperhatikan analisis berbasis frasa. Padahal, sebaiknya tidak hanya memperhatikan analisis *single-term* saja, akan tetapi perlu diperhatikan juga analisis frasa dari suatu dokumen. Dengan analisis frasa, kesamaan antar dokumen akan dihitung berdasarkan *matching phrase*.

Pada makalah ini akan digunakan *Document Index Graph* (DIG) sebagai metode klusterisasi yang tidak hanya memperhitungkan term yang terdapat dalam suatu dokumen, akan tetapi juga memperhitungkan ada atau tidaknya *matching phrase* yang terbentuk antara dokumen tersebut dengan dokumen lainnya. Selain itu DIG dianggap mampu menangani kluster yang *overlapping* dengan cara *incremental clustering*. *Overlapping* artinya mampu menghasilkan kluster yang tumpang tindih.

Penggunaan DIG sebagai algoritma klusterisasi, diharapkan menghasilkan aplikasi yang dapat bermanfaat dalam teknologi informasi, misalnya aplikasi pengelompokan dokumen berita berbasis *web* yang mempermudah pencarian informasi mengenai suatu kategori atau kejadian tertentu.

Tujuan yang ingin dicapai dalam makalah ini adalah:

1. Mengimplementasikan algoritma DIG yang mampu mendeteksi kesamaan berbasis frasa dan menangani *overlap clustering.*
2. Menganalisa performansi algoritma DIG berdasarkan nilai *f-measure* dan *entropy* dari hasil klusterisasi.

2. DASAR TEORI

2.1 Text Mining

Seperti halnya data mining, *text mining* adalah proses penemuan akan informasi atau *trend* baru yang sebelumnya tidak terungkap dengan memproses dan menganalisa data dalam jumlah besar. Dalam menganalisa sebagian atau keseluruhan *unstructured text*, *text mining* mencoba untuk mengasosiasikan satu bagian teks dengan yang lainnya berdasarkan aturan-aturan tertentu. Hasil yang di harapkan adalah informasi baru atau *insight* yang tidak terungkap jelas sebelumnya.

Banyaknya data yang ada, terkadang membuat kita kesulitan untuk menemukan substansi informasi dari data itu sendiri. Banyak hal telah dilakukan untuk menangani data sedemikian sehingga *user* lebih mudah untuk menemukan informasi yang mereka butuhkan dengan tepat dan dalam waktu yang efisien.

Klusterisasi merupakan salah satu metode untuk pengelompokan dokumen dengan menemukan keterkaitan antar dokumen. Keterkaitan antar dokumen dapat diketahui berdasarkan jarak centroid masing-masing data yang direpresentasikan dengan titik-titik pada bidang koordinat. Untuk mendapatkan kluster yang koheren, maka salah satu langkah yangdiambil adalah memaksimalkan jarak *intracluster* dan meminimalkan jarak *intercluster*. Keterkaitan antar dokumen dapat dilihat dari kesamaan kata maupun kesamaan frasa.

2.2 Frasa

Menurut kamus bahasa Indonesia, frasa atau frase adalah sebuah istilah linguistik. Lebih tepatnya, frase merupakan satuan linguistik yang lebih besar dari kata dan lebih kecil dari klausa dan kalimat. Frase adalah kumpulan kata nonpredikatif. Artinya frase tidak memiliki predikat dalam strukturnya. Beberapa contoh frase :

- ayam hitam saya
- ayam hitam
- ayam saya
- rumah besar itu
- rumah besar putih itu
- rumah besar di atas puncak gunung itu

Dalam makalah ini, yang dimaksud dengan frasa adalah urutan kata yang terdapat dalam sebuah kalimat pada sebuah dokumen.

2.3 Analisis Struktur Dokumen

Dokumen *web* biasanya berbentuk dokumen semistruktur. *Tag-tag* HTML digunakan untuk menunjukkan *lay out* atau bagian-bagian dari sebuah dokumen. Idenya, beberapa bagian dari sebuah dokumen mempunyai nilai informasi yang lebih tinggi dari bagian lainnya. Oleh karena itu bagian tersebut mempunyai tingkat kepentingan yang berbeda sesuai posisinya dalam sebuah dokumen [7]. Misalnya sebuah dokumen teks, yang terdiri dari judul dan isi dokumen. Dari dokumen tersebut mempunyai beberapa kata yang terletak pada bagian judul dan pada bagian isi dokumen. Tentunya kata-kata yang terletak pada bagian judul akan memiliki nilai kepentingan yang lebih tinggi daripada kata-kata yang berada pada bagian isi dokumen.

Biasanya, tingkat kepentingan sebuah kata yang terdapat di dalam suatu dokumen dibagi menjadi tiga tingkat, yaitu; tinggi, sedang, dan rendah. Contoh bagian dari dokumen yang mempunyai tingkat kepentingan tinggi adalah judul. Contoh bagian dokumen yang mempunyai tingkat kepentingan sedang adalah kata-kata yang dicetak tebal, kata-kata yang dicetak miring, atau kata-kata yang diberi warna. Sedangkan tingkat kepentingan rendah biasanya isi dokumen yang tidak termasuk ke dalam tingkat kepentingan tinggi maupun tingkat kepentingan sedang.

2.4 *Document Index Graph (DIG)*

DIG merupakan algoritma pembangun graf. Graf yang dibangun merupakan graf berarah, dimana arahnya menunjukkan struktur kalimat. Graf yang dibangun merupakan komponen dari :

1. *Node*
   Node berisi kata unik dari setiap kalimat dalam dokumen.
2. *Edge*
   Merupakan penghubung antar*node*. Pada *edge* terdapat informasi berupa nomor *edge*, posisi kata tersebut dalam kalimat dan dalam dokumen.
3. *Path*
   Node pada graf berisi informasi tentang kata unik dalam sebuah dokumen. Jalur atau *path* yang dibentuk oleh *node* dan *edge* merupakan representasi dari sebuah kalimat tertentu.

DIG incremental construction and phrase matching

Require: G_{i-1}: cumulative graph up to document d_{i-1} or G_0 if no documents were processed previously
$d_i \leftarrow$ Next Document
$M \leftarrow$Empty List {M is a list of matching phrases from previous documents}
for each sentences s_{ij} in d_i **do**
$v_1 \leftarrow t_{ij1}$ {first word in s_{ij}}
if v_1 is not in G_{i-1} **then**
Add v_1 to G_{i-1}
end if
for each term $t_{ijk} \in s_{ij}$, $k = 2,\ldots, l_{ij}$ **do**
$v_k \leftarrow t_{ijk}$; $v_{k-1} \leftarrow t_{ij(k-1)}$; $e_k = (v_{k-1}, v_k)$
if v_k is not in G_{i-1} then
Add v_k to G_{i-1}
end if
if e_k is an edge in G_{i-1} then
Retrieve a list of document entries from v_{k-1} document table that have a sentence on the edge e_k
Extend previous matching phrases in M for phrases that continue along edge e_k
Add new matching phrases to M
else
Add edge e_k to G_{i-1}
end if
Update sentence path in nodes v_{k-1} and v_k
end for
end for
$G_i \leftarrow G_{i-1}$
Output matching phrases list M

2.5 Kesamaan Dokumen

Nilai kesamaan dokumen dapat dihitung melalui beberapa pendekatan :

2.5.1 *Single Term*

Single term atau kesamaan dokumen berbasis kata, merupakan nilai kesamaan dokumen yang dilihat berdasarkan term-term yang berada di antara dua dokumen yang sedang dibandingkan. Metode *Cosine Based Similarity* dapat digunakan untuk memperoleh nilai kesamaan dokumen berbasis kata. Dengan mengukur dua vektor berdimensi n dengan menemukan sudut diantara keduanya. Untuk *text-matching* , atribut yang biasa dipakai adalah vektor TF-IDF. Ukuran kesamaan dokumen d_1 dengan dokumen d_2 dapat dihitung dengan persamaan :

$$simt(d1, d2) = \cos(d1, d2) = \frac{d1 d2}{|d1||d2|} \quad (2.1)$$

TF atau term frequency merupakan banyaknya term dalam sebuah dokumen. Pembobotan TF diperoleh dari perhitungan dengan persamaan :

$$tf = \frac{f}{m}, \quad f > 0 \quad (2.2)$$

f = frekuensi term dalam sebuah dokumen
m = frekuensi maksimum dari suatu term yang terdapat dalam sebuah dokumen

Sedangkan IDF atau Inverse Document Frequency merupakan banyaknya term tertentu dalam keseluruhan dokumen. Pembobotan IDF dapat dihitung dengan rumus :

$$idfj = \log_2\left(\frac{n}{n_j}\right) + 1 \quad n_j > 0 \quad (2.3)$$

n = jumlah seluruh dokumen
n_j = jumlah dokumen yang mempunyai term j

2.5.2 *Phrase Based Similarity*

Metode ini akan menggunakan frasa sebagai tolok ukur kesamaan dokumen. Persamaan dokumen yang diukur berdasarkan term dianggap belum memberikan hasil yang terbaik[7]. Dengan memperhatikan urutan dari beberapa kata yang terdapat di antara dua dokumen yang sedang dibandingkan diharapkan dapat meningkatkan nilai akurasi pengelompokan dokumen.

Ukuran kesamaan dokumen dihitung berdasarkan *shared phrase* pada masing-masing pasangan dokumen.

Faktor –faktor *shared phrase* dalam menentukan kesamaan dokumen :

a. jumlah *matching phrase*,
b. panjang *matching phrase,*
c. frekuensi *matching phrase* di kedua dokumen
d. level signifikan(*weight*) dari *matching phrase* di kedua dokumen tersebut.

Kesamaan berbasis frasa antara 2 dokumen, d1 dan d2 dapat dihitung dengan persamaan :

$$simp(d_1, d_2) = \frac{\sqrt{\sum_{i=1}^{f} [g(li).(f1i\, w1i + f2i\, w2i)]^2}}{\sum_j |s1j| . w1j + \sum_k s2k| . w2k} \quad (2.4)$$

$$g(li) = (li/|si|)^y \quad (2.5)$$

2.5.3 Gabungan antara *Single Term* dan *Phrase Based Similarity*

Kesamaan dokumen akhir dihitung dari kombinasi antara kesamaan berbasis kata dengan kesamaaan berbasis frasa dengan persamaan berikut:

$$sim(d_1, d_2) = \alpha . simp(d_1, d_2) + (1 - \alpha) . simt(d_1, d_2) \quad (2.6)$$

2.6 *Similarity Histogram Clustering*

Klusterisasi merupakan salah satu metode untuk pengelompokan dokumen dengan menemukan keterkaitan antardokumen. Keterkaitan antar-dokumen dapat diketahui berdasarkan jarak centroid masing-masing data yang direpresentasikan dengan titik-titik pada bidang koordinat. Untuk mendapatkan kluster yang koheren, maka salah satu langkah yangdiambil adalah memaksimalkan jarak intracluster dan meminimalkan jarak interclusternya.

Histogram merupakan representasi statistik dari pasangan dokumen yang terdapat dalam sebuah kluster. Histogram yang terbentuk akan digunakan untuk menghitung histogram ratio dengan rumus :

$$HR_c = \frac{\sum_{i=r}^{g} hi}{\sum_{j=1}^{g} hj} \quad (2.7)$$

$$T = [S_T.B] \quad (2.8)$$

HR_c = *histogram ratio* untuk kluster c

S_T = *similarity threshold*
T = jumlah batang histogram yang sesuai dengan *similarity threshold*

Similarity Histogram-based Incremental Document Clustering

```
L ← Empty List {Cluster List}
for each document d do
    for each cluster c in L do
        HR_old = HR_c
        Simulate adding d to c
        HR_new = HR_c
        if (HR_new≥HR_old) OR ((HR_new>HR_min) AND (HR_old - HR_new<ε)) then
            Add d to c
        end if
    end for
    if d was not added to any cluster then
        Create a new cluster c
        ADD d to c
        ADD c to L
    end if
end for
```

2.7 Evaluasi Klusterisasi

Evaluasi ini dilakukan untuk mengetahui kinerja dari algoritma klusterisasi dalam tahap uji coba. Pengukuran ini didasarkan pada 2 ukuran kualitas kluster yang biasa digunakan dalam literatur pengukuran klusterisasi dokumen.

2.7.1 *F-measure*

$$R = \frac{Nij}{Ni} \quad (2.9)$$

$$P(Ki, Cj) = \frac{nij}{nj} \quad (2.10)$$

Nij = jumlah anggota kelas ke-i pada kluster ke-j
Ni = jumlah anggota kelas ke-i
Nj = jumlah anggota kluster ke-j

F-measure dari kluster Cj dan kelas Ki dapat didefinisikan sebagai :

$$F(i) = \frac{2*PR}{R+P} \quad (2.11)$$

Untuk *histogram clustering*, *F-measure* dari setiap kelasnya merupakan rata-rata nilai *F-measure* dari tiap kelas pada keseluruhan kluster yang terbentuk.

$$F(C) = \frac{\sum_i (|i| \times F(i))}{\sum_i |i|} \quad (2.12)$$

|i| = jumlah anggota masing-masing kelas ke-i

2.7.2 *Entropy*

Entropy mengukur kemurnian dari kluster yang dihasilkan dengan memperhatikan pada kategori yang ada. Nilai *Entropy* yang lebih kecil menghasilkan kluster yang lebih bagus kualitasnya.

$$Ej = -\sum_i pij \log(pij) \quad (2.13)$$

Total *entropy* dihitung sebagai jumlah dari nilai *entropy* tiap-tiap kluster yang terbentuk.

$$Ec = \sum_{j=1}^{m} \frac{Nj}{N} X\ Ej \quad (2.14)$$

Nj = jumlah dokumen yang diklusterkan dalam satu kategori

N = total jumlah dokumen

p_{ij} = peluang dokumen kluster j masuk ke kelas i

3. ANALISIS DAN PERANCANGAN SISTEM

Perancangan sistem memakai konsep *Object Oriented*. Sistem di implementasikan dalam bentuk aplikasi desktop dengan menggunakan nahasa pemprograman Java.

4. PENGUJIAN

4.1 Dataset yang digunakan

4.1.1 Dataset *Single-label*

Dataset ini merupakan dataset *IndonesianTREC-like Corpus* yang berisikan kumpulan artikel-artikel yang berasal dari media surat kabar kompas (www.kompas.com), dataset ini menggunakan 180 artikel dengan 6 kategori.

Kategori dan komposisi dokumen untuk 180 artikel, di antaranya:

1. "Kecelakaan pesawat udara Indonesia" yang berjumlah 22 artikel.
2. "Situasi banjir Jakarta" yang berjumlah 40 artikel.
3. "Duta besar Indonesia" yang berjumlah 30 artikel.
4. "Nama suami Megawati" yang berjumlah 28 artikel.
5. "Nama bos Manchester United" yang berjumlah 25 artikel.
6. "Nilai tukar rupiah terhadap Dollar AS" yang berjumlah 35 artikel.

4.1.2 Dataset *Multilabel*

Dataset ini merupakan dataset yang berisikan kumpulan artikel-artikel yang berasal dari media surat kabar online. Dataset ini menggunakan 90 artikel dengan 3 kategori.

Kategori dan komposisi dokumen untuk 90 artikel, di antaranya:

1. " Hiburan" yang berjumlah 90 artikel.
2. " Hukum" yang berjumlah 30 artikel.
3. " Politik" yang berjumlah 60 artikel.

4.2 Pengujian Sistem

Dari implementasi terhadap perangkat lunak pada Bab III dihasilkan aplikasi klusterisasi dengan metode DIG. Dengan aplikasi ini dilakukan simulasi dan pengujian terhadap dokumen uji untuk mengetahui perbandingan performansi kluster yang dihasilkan dengan pengukuran kesamaan berbasis frasa dan performansi kluster yang dihasilkan tanpa pengukuran kesamaan berbasis frasa. Inputan yang

digunakan merupakan dokumen berekstensi .txt. Hasil keluaran dari sistem ini berupa file berekstensi .csv yang berisi id_dokumen, nilai_histogram_ratio, dan status_penerimaan_kluster. Aspek yang akan digunakan untuk menilai performansi kluster adalah *f-measure* dan *entropy.*

Nilai *fragmentation factor* yang digunakan adalah 1.2, karena berdasarkan pengujian sebelumnya, nilai tersebut dapat memberikan hasil terbaik. Dalam pengujian terhadap sistem, dilakukan hal-hal berikut ini :

1. Analisis performansi algoritma DIG .

   Pengujian dilakukan dengan menggunakan dataset single-label. Kesamaan dokumen dihitung menggunakan sistem untuk beberapa similarity blend factor dan similarity threshold kemudian dilakukan proses klusterisasi yang akan menghasilkan nilai histogram ratio. Nilai similarity blend factor yang digunakan antara 0 sampai dengan 1, sedangkan similarity threshold yang diujicobakan bernilai 0.01 sampai dengan 1. Hasil klusterisasi dianalisis berdasarkan f-measure dan entropy kemudian dilakukan perbandingan hasil klusterisasi pada dataset yang menggunakan kesamaan berbasis kata dan data set yang menggunakan kesamaan berbasis kata dan berbasis frasa.
2. Analisis overlap clustering

   Pengujian dilakukan dengan menggunakan dokumen *multilabel* untuk melihat pengaruhnya terhadap nilai *histogram ratio*. Dokumen yang dapat memperbaiki *histogram ratio* dapat dikelompokkan dengan kluster tersebut. Sedangkan dokumen yang menurunkan *histogram ratio* pada semua kluster yang ada akan dikelompokkan dalam kluster tersendiri.

5 KESIMPULAN DAN SARAN

5.1 Kesimpulan

1. Algoritma DIG dapat diimplementasikan untuk mendeteksi kesamaan berbasis frasa dan menangani *overlap clustering.*
2. Kesamaan berbasis frasa dapat memperbaiki performansi kluster berdasarkan pengukuran *f-measure* dan *entropy*. Tetapi kesamaan berbasis frasa tidak selalu dapat memperbaiki kualitas atau performansi kluster berdasarkan *f-measure* dan *entropy*. Ada beberapa titik di mana kesamaan berbasis frasa justru dapat mengurangi nilai performansi, oleh karena itu perlu dicari titik optimal *similarity blend factor* dan *similarity threshold.*
3. Pada kasus dataset *single-label* pada makalah ini, hasil optimal dari klusterisasi akan diperoleh pada pemilihan kombinasi *similarity blend factor* 0.7 sampai dengan 0.9 dan *similarity threshold* 0.01 sampai dengan 0.6.
4. Pada kasus dataset *multilabel* pada makalah ini, hasil optimal dari klusterisasi akan diperoleh pada pemilihan kombinasi *similarity blend factor* 0 sampai dengan 0.8 dan *similarity threshold* 0.01 sampai dengan 0.1

5.2 Saran

Saran terhadap pengembangan yang akan dilakukan terhadap makalah ini adalah :

1. Perlu melakukan analisis struktur dokumen yang lebih lengkap; bagian dokumen yang mempunyai tingkat kepentingan tinggi, sedang, dan rendah. Hal ini diperlukan untuk perhitungan kesamaan berbasis frasa yang lebih akurat.
2. Perlu melakukan analisis terhadap makna frasa.

6 DAFTAR PUSTAKA

Adiwijaya, Igg. (2006). *Texi Mining dan Knowledge Discovery.* Komunitas *Data mining* Indonesia & Soft-omputing Indonesia.

Arifin, Agus Z dan Setiono, Ari N. 1998. "Klasifikasi Dokumen Berita Kejadian Berbahasa Indonesia dengan Algoritma Single Pass Clustering"

Beil, Florian, et.al. 2002. " Frequent Term-Based Text Clustering "

Bourigault, Didier dan Jacquemin, Christian. 1999. "Term extraction + Term clustering : An Integrated Platform for Computer-Aided Terminology"

Feldman, Ronen, et.al. 1998. " Knowledge Management: A Text Mining Approach"

Flesca, Sergio, et.al.2002. " Detecting Structural Similarities between XML Documents"

Hammouda, Khaled M and Kamel, Mohamed S. 2004. "Efficient Phrase-Based Document Indexing for Web Document Clustering".

Hung Chim and Xiaotie Deng.2007."A New Sufix Tree Similarity Measure for Document Clustering"

Isaacs, Jeffrey and Aslam, Javed. 1999. "Investigating Measures for Pairwise Document Similarity".

Steinbach, Michael, et.al.2001."A Comparison Document Clustering Techniques"

Tala, Fadillah Z .2002. " A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia" 2003

Wibisono, Yudi dan Khodra, Masayu L. 2006. "Clustering Berita Berbahasa Indonesia"