ANALISIS DAN IMPLEMENTASI ALGORITMA OVERLAPPING COVER COEFFICIENT-BASED CLUSTERING METHOD (OC3M) PADA DOKUMEN TEKS BERBAHASA INDONESIA

ANALYSIS AND IMPLEMENTATION OF OVERLAPPING COVER COEFFICIENT-BASED CLUSTERING METHOD (OC3M) ALGORITHM ON INDONESIAN TEXT DOCUMENT

Gusti Ngurah Diva A[1], Shaufiah, ST.,MT.[2], Veronikha Effendy, ST.,MT.[3]

[1,2,3]Prodi S1 Teknik Informatika, Fakultas Teknik Informatika, Universitas Telkom
[1]diva.adwitya@gmail.com, [2]shaufiah@gmail.com, [3]veffendy@telkomuniversity.ac.id

Abstrak

Dengan bertambah pesatnya informasi/dokumen yang beredar di internet sehingga memungkinkan untuk suatu dokumen dapat dikelompokkan ke dalam dua atau lebih kategori sekaligus. Oleh karena itu dibutuhkan suatu metode untuk mengelompokkan dokumen-dokumen tersebut ke dalam dua atau lebih kategori sekaligus.

Overlapping Cover Coefficient Clustering Method (OC3M) adalah suatu metode pengelompokan dokumen dengan model probabilitik, kesamaan *term*, dan *seed* dokumen sebagai inisialisasi awal dari pembentukan *cluster*. Pada metode ini diterapkan sifat *overlap*, yaitu kondisi dimana dokumen dapat menempati lebih dari satu *cluster*.

Pengujian yang dilakukan pada tugas akhir ini dalam mengelompokkan dokumen dengan algoritma OC3M yaitu menganalisis *cluster* yang dihasilkan berdasarkan nilai *Silhouette Coefficient*-nya serta menganalisis hal-hal yang mempengaruhi kualitas *cluster* yang terbentuk. Kualitas *cluster* yang terbentuk dipengaruhi oleh banyaknya dokumen yang digunakan, tipe dokumen, kemiripan dokumen dengan pusat *cluster*, dan juga dipengaruhi oleh *overlapping coefficient* yaitu parameter yang menentukan banyaknya suatu dokumen yang *similar* dapat dikelompokkan ke dalam *cluster* yang berbeda. Dari hasil percobaan, kualitas cluster yang terbentuk dengan menggunakan algoritma OC3M memiliki *kualitas* yang cukup baik, ini di tunjukkan dengan nilai *silhouette coefficient* yang bernilai positif.

Kata kunci : OC3M, Overlap, Overlaping, Clustering, Cluster

Abstract

With the rapid increase of information / documents circulating on the internet making it possible for a document can be grouped into two or more categories. For that it needed a method to be able to classify documents in which a document can be grouped into two or more categories.

Overlapping Clustering Cover Coefficient Method (OC3M) is a method of grouping documents with probabilitic models, terms similarity, and the seed documents as early initialization of cluster formation. In this method, it is applied an overlaping properties, which is causing a document to be able occupy more than one cluster.

A tests conducted in this thesis in clustering a document with OC3M algorithm is by analyzing a produced clusters using Silhouette Coefficient value and analyzing the things that affect the quality of the formed cluster. The quality of clusters is influenced by the number of documents used, type of document, the document similarity with the cluster's center, and also influenced by the overlapping coefficient which is a parameter that determines a similar documents could be grouped into a different clusters. The result of the experiment in clustering using OC3M algorithm is quite good, it's shown in the silhouette coefficient value, which is a positive value.

Keywords : OC3M, Overlap, Overlaping, Clustering, Cluster

Pendahuluan

1.1. Latar Belakang

Dengan bertambah besarnya jumlah dokumen yang beredar di internet, terdapat beberapa dokumen yang dapat di kelompokkan ke dalam beberapa kelompok berbeda sekaligus. Salah satu contohnya pada *dataset 20-Newsgroup* yang digunakan pada klasifikasi teks dan *clustering*, sejumlah artikel asli ternyata di posting ke beberapa *newsgroup* berbeda sekaligus; data itu kemudian di manipulasi untuk menghasilkan kategori *disjoint*. Idealnya algoritma *clustering* yang diterapkan pada data ini akan mengijinkan artikel untuk menempati beberapa *newsgroup* berbeda sekaligus [2]. Contoh lainnya pada koleksi dokumen berita, berita "Teuku Reza Tertangkap Saat Pesta Ganja", dapat di kelompokkan ke dalam dua kelompok berbeda yaitu kelompok berita kriminal dan

kelompok berita hiburan. Sayangnya algoritma *clustering* yang kebanyakan beredar tidak dapat mengatasi masalah *overlap* tersebut, algoritma tersebut hanya dapat mengelompokkan ke dalam satu kelompok tertentu, tidak bisa kedalam beberapa kelompok sekaligus. Oleh karena itu diperlukan suatu cara/metode untuk dapat mengatasi hal tersebut.

Overlapping clustering adalah suatu metode yang digunakan untuk dapat mengelompokkan dokumen ke dalam beberapa kelompok berbeda sekaligus. Pengelompokan ini dilakukan dengan melihat nilai similaritas yang dimiliki dokumen dan pusat *cluster*. Ketika nilai similaritas dari suatu dokumen terhadap suatu *cluster* lebih dari nol, maka dokumen tersebut dapat dikelompokkan dengan *cluster* tersebut.

Pada penelitian yang akan dilakukan dalam tugas akhir ini untuk mengatasi permasalahan *overlap* di atas adalah dengan menggunakan algoritma *Overlapping Cover Coefficient Clustering Method* (OC3M) pada dokumen teks khususnya pada dokumen berbahasa Indonesia, karena *clustering* pada dokumen bahasa Indonesia masih tergolong sedikit. Algoritma OC3M ini merupakan varian dari *Cover Coefficient Clustering Method* (C3M) dimana pada algoritma C3M sudah cukup terbukti dalam menghasilkan *cluster* dengan kualitas yang bagus, memiliki waktu pemrosesan yang cepat dan dengan menggunakan konsep *Cover Coefficient* sehingga tanpa perlu *input*-an *user* dalam menentukan jumlah *cluster* yang harus dibentuk. Algoritma ini akan menentukan berapa cluster yang akan dibentuk dengan menggunakan model probabilitik dan kesamaan *term* antar dokumen serta menggunakan *seed power* untuk sebagai pola awal *cluster*.

1.2. Perumusan Masalah

Rumusan masalah yang diangkat pada tugas akhir ini adalah:

a. Bagaimana implementasi *clustering* dengan menggunakan *overlapping clustering*, khususnya dengan algoritma OC3M pada dokumen teks berbahasa Indonesia?
b. Bagaimana pengaruh dari parameter-parameter OC3M yaitu penggunaan TF/Non-TF, *threshold*, *overlapping coefficient* terhadap hasil *clustering*?
c. Bagaimana kualitas *cluster* yang terbentuk dari algoritma OC3M bila dihitung berdasarkan nilai *Silhouette Coefficient*?

1.3. Batasan Masalah

Berikut merupakan batasan masalah dalam Tugas Akhir ini:

a. Koleksi dokumen yang digunakan adalah artikel berita yang diambil dari website kompas (http://www.kompas.com) dari tanggal 1 Oktober 2014 hingga 31 Maret 2014 yang berjumlah 2690 dokumen dan dokumen abstrak TA Universitas Telkom Fakultas Teknik Informati yang berjumlah 200 dokumen.
b. Penelitian pada Tugas Akhir ini hanya terfokus pada proses clustering dokumen dengan satu metode saja yaitu OC3M (*Overlapping Cover Coefficient Clustering Method*).
c. Algoritma OC3M (*Overlapping Cover Coefficient Clustering Method*) tidak menangani pelabelan sehingga tidak ada penamaan dalam cluster yang terbentuk.

1.4. Tujuan

Tujuan dari pembuatan tugas akhir ini adalah:

a. Melakukan implementasi clustering dengan menggunakan algoritma OC3M pada dokumen teks berbahasa Indonesia.
b. Menganalisa hasil clustering yang terjadi dan efisiensi algoritma OC3M pada proses clustering dokumen dengan mengunakan parameter yang berbeda.
c. Mengukur dan menganalisa cluster yang terbentuk dengan algoritma OC3M berdasarkan nilai *Silhouette Coefficient*-nya.

1.5. Metode Penelitian

Metode penyelesaian masalah yg dilakukan dalam menyelesaikan tugas akhir ini adalah:

a. Studi literatur
   Mencari dan mengumpulkan informasi dari buku maupun artikel dan paper-paper serta memahaminya sehingga diperoleh dasar teori yang dapat digunakan untuk menyusun tugas akhir.
b. Mengumpulkan document collection
   Mengumpulkan data berupa dokumen teks yang dibutuhkan untuk keperluan proses implentasi dan pengujian algoritma *Overlapping* C3M.

c. Perancangan Sistem
Melakukan perancangan terhadap perangkat lunak sistem *clustering* yang akan dibangun dengan menggunakan algoritma OC3M. *Input*-an berupa dokumen, kemudian dilakukan *preprocessing* dan keluaran adalah berupa *cluster* dari dokumen *input*-an.

d. Implementasi
Melakukan pembangunan terhadap sistem yang telah dirancang sebelumnya. Sistem akan dibuat dengan menggunakan bahasa pemrograman *java*, *software* yang digunakan untuk pembuatan sistem ini adalah *Netbeans*.

e. Pengujian sistem dan analisa hasil
Pada tahap ini dilakukan pengujian terhadap sistem yang telah dibuat. Dalam sistem ini pengujian dilakukan bertujuan unuk menganalisa parameter-parameter algoritma OC3M terhadap hasil *clustering* dan menganalisa performansi algortima OC3M dalam menghasilkan *cluster*.

f. Penyusunan laporan
Pembuatan laporan tugas akhir yang mendokumentasi semua tahap kegiatan dan hasil dalam tugas akhir ini.

2. Dasar Teori

2.1 *Overlapping Clustering*

Dalam suatu kondisi tertentu, terdapat suatu dokumen yang dapat dikategorikan pada banyak kategori seperti contohnya pada kumpulan dokumen sinopsis film, dimana satu dokumen dapat dikategorikan sebagai film aksi, horror, dan drama. *Overlapping Clustering* merupakan suatu metode *clustering* dimana tiap objeknya dapat menempati lebih dari satu *cluster*. Dengan berlakunya sifat *overlapping*, sehingga untuk contoh kumpulan dokumen diatas dapat teratasi karena dalam satu dokumen dapat menempati lebih dari satu cluster. Salah satu algoritma yang menerapkan overlapping clustering adalah *Overlapping Cover Coefficient Clustering Method* (OC3M).

2.2 *Overlapping Cover Coefficient-Based Clustering Methodology* (OC3M)

OC3M diperkenalkan oleh Xinpei Shu untuk klasterisasi dokumen teks yang bersifat *overlap* [12]. Konsep dasar dari algoritma ini, *Cover Coefficient* (CC) menyediakan estimasi rata-rata jumlah *cluster* sehingga *user* tidak perlu meng-*input*-kan jumlah *cluster* (*no user interaction and asumption*) dalam dokumen *database* dan menghubungkan pengindeksan *database* (dokumen) serta *clustering*. Konsep dasar dari algoritma CC menentukan estimasi rata-rata jumlah *cluster* dalam sebuah dokumen *database*. Konsep CC juga digunakan untuk meng-identifikasi *cluster seed* melalui perhitungan *seedpower*.

Algoritma OC3M merupakan metode *clustering* yang memanfaatkan *seed document* sebagai inisialisasi/pola awal untuk pembentukan *cluster*. Dalam pembentukan *seed document*, algoritma ini menggunakan model probabilistik dalam menggambarkan keterkaitan antar dokumen ataupun antar *term*-nya.

Pembentukan *cluster* dengan algoritma OC3M hampir sama dengan algoritma akarnya yaitu C3M. Pada algoritma C3M terdapat 2 fase dalam pembentukan *cluster*-nya, yaitu fase seleksi *cluster seed* (*seed power*) dan fase konstruksi *cluster* [12].

2.2.1 Fase Seleksi Cluster Seed

Pada fase ini terdapat 2 hal yang yang dilakukan dengan menggunakan *cover coefficient* yaitu estimasi jumlah *cluster* dan menentukan nilai *seed power* dokumen.

Dalam menentukan jumlah *cluster*, hal pertama yang dilakukan adalah membentuk vektor D-Matriks. D-Matriks ini merupakan pemetaan dokumen dengan *term-term* yang ada pada tiap dokumen. D-Matriks dapat digambarkan seperti berikut, misalkan terdapat dokumen d_i dan d_j, serta term t_k. Langkah pertama yang dilakukan adalah memilih secara acak *term* t_k dalam dokumen d_i, kemudian langkah kedua adalah memilih *term* t_k terpilih dari dokumen di tersebut di dalam dokumen d_j. Langkah-langkah tersebut diulang untuk semua *term* yang ada dan untuk semua dokumen.

Setelah melakukan pemetaan dengan D-Matriks, selanjutnya yaitu pembentukan C-Matriks. C-Matriks merupakan suatu nilai yang menunjukkan keterhubungan antara dokumen dengan *term*. Untuk membangun C-Matriks yang berukuran m x m, diperlukan nilai c_{ij} dari D-Matriks yang terbentuk. Perhitungan nilai c_{ij} menggunakan persamaan 2.1 berikut [4]:

$$c_{ij} = \alpha_i \sum_{k=1}^{t} (d_{ik} \times \beta_k \times d_{jk}) \qquad 1 \le i, j \le m \qquad (2.1)$$

Jika terdapat dokumen yang tidak mempunyai *term* yang sama maka mereka tidak akan meng-*cover* satu sama lain, sehingga c_{ij} dan c_{ii} akan bernilai nol. Dan c_{ii} akan bernilai 1 jika semua *term* yang dimiliki c_{ii} tidak dimiliki oleh dokumen lain. Karena tingginya nilai c_{ij} yang dihasilkan dari dokumen d_i yang mempunyai *term* yang sama dengan dokumen lainnya, maka c_{ij} dapat disebut dengan *decoupling coefficient* (δ_i) yang menggambarkan

perbedaan/*dissimilarity* d_i dengan semua dokumen dan *coefficient* yang menggambarkan persamaan/similarity disebut *coupling coefficient* (φ_i). Dari nilai $\square$ yang dihasilkan dapat ditentukan jumlah cluster yang terbentuk dengan persamaan 2.2 [4]:

$$\square = \sum_{\square=1}^{\square} \square \text{ dimana } 1 \leq \square \leq \min(\square\square \quad (2.2)$$

Setelah terbentuknya C-Matriks dan melakukan penghitungan jumlah cluster, dilanjutkan dengan menghitung seed power tiap dokumen. Seed power ini akan menjadi pusat dari tiap cluster yang terbentuk. Nilai seed power dari tiap dokumen yang paling tinggi akan menjadi seed dokumen.

$$\square\square = \square \times \varphi_{\square} \times \sum_{\square=1}^{\square} \square \quad (2.4)$$

Untuk menghitung nilai *decoupling coefficient* (δ_i) dan *coupling coefficient* (φ_i) dilakukan dengang menggunakan persamaan 2.5 dan 2.6 berikut [4]:

$$\square = \square\square \quad (2.5)$$

$$\varphi_{\square} = 1 - \square \quad (2.6)$$

2.2.2 Fase Konstruksi *Cluster*

Fase ini akan menempatkan dokumen-dokumen *nonseed* kedalam dokumen *seed*. Dokumen *nonseed* akan ditempatkan berdasarkan nilai persamaan/*similarity*-nya dengan dokumen *seed*.

Berbeda dengan C3M, pada OC3M, dalam fase ini satu dokumen dapat menempati lebih dari satu *cluster*. Dengan berlakunya sifat *overlapping* yang dipengaruhi oleh *overlapping coefficient* (*k*) yang digunakan untuk mengatur jumlah dari tiap dokumen *nonseed* untuk menempati *cluster* [5]. Sehingga untuk dokumen nonseed yang nilai similaritasnya tidak sama dengan nol dapat menempati *cluster* yang sebanyak *overlapping coefficient* (*k*) yang telah ditentukan sebelumnya. Misalnya, jika *overlapping coefficient* (*k*) bernilai dua, maka tiap dokumen *nonseed* akan menempati dua *cluster* berbeda sesuai dengan nilai similaritasnya.

Pada *overlapping* ini, *threshold* digunakan untuk mengontrol dokumen yang akan menempati *cluster* selanjutnya [5]. Nilai dari *threshold* (*h*) itu sendiri berkisaran antara 1 hingga 0 dan dapat di definisikan sebagai berikut (1 > h > 0) [5]. Nilai *threshold* ini akan digunakan ketika dokumen nonseed akan menempati *cluster* selanjutnya. Berdasarkan nilai c_{ij} dokumen *seed* pada *cluster* pertama maka untuk d_k jika nilai $\square \geq h \times \square$ maka d_k dapat dimasukkan ke dalam *cluster* berikutnya [5].

2.3 Evaluasi Cluster

Terdapat dua metode umum untuk mengukur tingkat keberhasilan hasil clustering yaitu *internal* dan *external measure*. *Internal measure* membandingkan *cluster-cluster* yang dihasilkan tanpa adanya informasi atau *knowledge* atas kelas-kelas awal sebelumnya. Sedangkan *external measure* mengevaluasi *cluster-cluster* yang dihasilkan dengan kelas-kelas yang sudah ditentukan sebelumnya [8]. Pada Tugas Akhir ini akan digunakan metode *internal measure* yaitu *silhouette coeffiecient* (SC) sebagai metode pengukuran. Pada metode ini pengukuran kualitas akan terbentuknya *cluster* diukur melalui dua pendekatan pengukuran, yaitu mengukur jarak kedekatan antar dokumen dalam satu *cluster*-nya sendiri atau dapat disebut juga *Cohesion* dan perhitungan nilai jarak minimum kedekatan dokumen antar *cluster* yang berbeda atau *Separation* [8].

Langkah – langkah dalam perhitungannya adalah sebagai berikut [7]:

a. Untuk data objek ke-i dalam cluster A, maka nilai a(i) adalah rata-rata jarak objek ke-i ke semua data objek dalam *cluster* yang sama dengannya.
b. Untuk data objek ke-i dalam *cluster* A, maka nilai b(i) adalah minimum rata-rata jarak objek ke-i ke semua data poin dalam cluster B (*cluster* yang berbeda dengan *cluster* A).

Penghitungan nilai *silhouette coefficient* untuk objek ke-i, dilakukan dengan menggunakan persamaan 2.7 berikut [7]:

$$\square\square = \frac{\square(\square\square) - \square(\square\square)}{\max\{\square\square, \square\square\}} \quad (2.7)$$

Nilai *Silhouette Coefficient* akan mencapai nilai maksimum 1 jika nilai a(i) = 0. Oleh karena itu nilai ini bernilai baik jika b(i) > a(i). Pada Tugas Akhir ini menggunakan rumus perhitungan jaraknya dengan rumus *cosine distance*.

Jika s(i) mendekati nilai -1, maka dapat dikatakan bahwa anggota *cluster* tersebut kurang tepat berada di *cluster* tersebut, sedangkan jika s(i) mendekati nilai 0, itu berarti data berada pada batas *cluster* atau dengan kata lain *overlapping cluster*. Dan nilai s(i) yang bernilai mendekati 1 yang berarti bahwa data sangat tepat berada pada cluster tersebut [7].

Persamaan 2.8 dan 2.9 berikut digunakan untuk menghitung *cosine* d*istance* dan *cosine similarity*-nya [10]:

$$\text{cosine distance} = 1 - \text{cosine similarity} \quad (2.8)$$

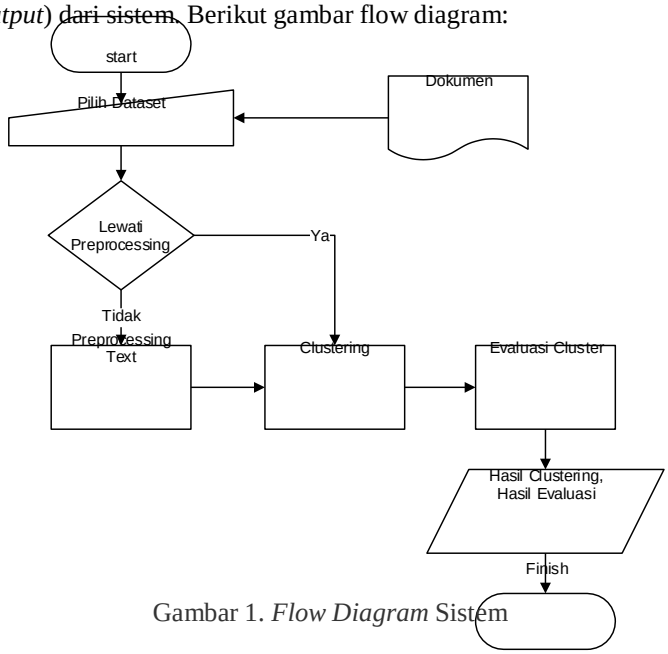
$$\text{cosine similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^{n} (A \times B)}{\sqrt{\sum_{i=1}^{n} (A)^2} \times \sqrt{\sum_{i=1}^{n} (B)^2}} \quad (2.9)$$

3. Perancangan Sistem

3.1 Deskripsi dan Gambaran Sistem

Sistem yang akan dibangun dalam Tugas Akhir ini berupa aplikasi *desktop* sederhana yang berfungsi untuk melakukan proses *clustering* sekumpulan dokumen dengan menggunakan metode OC3M. Data *input*-an berupa dokumen teks yang disimpan terlebih dahulu ke dalam *database*, kemudian dilakukan proses *preprocessing* dan dilanjutkan dengan proses *clustering*.

Berikut proses-proses yang dilakukan dalam sistem digambarkan dengan *flow diagram* agar lebih mudah dimengerti. *Flow diagram* dibawah ini menggambarkan secara urut proses-proses yang terjadi di dalam sistem, dari awal (*input*) sampai akhir (*output*) dari sistem. Berikut gambar flow diagram:



Gambar 1. *Flow Diagram* Sistem

3.1.1 Preprocessing Text

Sebelum melakukan *clustering*, untuk mendapatkan hasil *cluster* yang lebih baik dilakuan *preprocessing* terlebih dahulu. Proses ini merupakan proses pengolahan data awal dari sekumpulan dokumen sehingga di bentuk menjadi kata-kata (dasar) yang mewakili tiap dokumen agar mempermudah dalam analisis pencarian keterhubungan antar dokumen. Proses ini terdiri dari beberapa tahap antara lain sebagai berikut,

- *Tokenization*

Meliputi proses penghilangan tanda baca yang terdiri dari: “., ,, !, :, ;, ?, &, (,), [,],{,},_,-,%,$,@,<,>,#, *,‘, “,|, /” dan pengubahan setiap huruf menjadi bentuk *lowercase* serta menghilangkan semua angka.

- *Stoplist*

Penghapusan kata-kata umum (*stopword*) seperti ”ke, dari, pada, dan, yang, begini, kami, lebih, hanya, sesudah, beberapa, sedikit” dan sebagainya. Adapun daftar *stoplist* yang digunakan sebanyak lebih kurang 1300 kata.

- *Stemming*

Proses pencarian kata dasar (akar kata) dari dokumen yang sudah melalui proses *tokenization* dan *stoplist*. Disini digunakan algoritma Nazief & Adriani sebagai algoritma *stemming* untuk bahasa Indonesia karena algoritma tersebut cukup akurat dibandingkan yang lainnya dan dengan menggunakan kamus yang berjumlah besar sehingga dapat meningkatkan keakuratan algoritma ini. Adapun daftar kamus yang digunakan berjumlah lebih kurang 30.000 kata.

Karena proses dari *stemming* yang memerlukan cukup banyak waktu, sehingga pada sistem diterapkan sebuah fungsi untuk melewati proses *preprocessing* ini. Tentunya untuk mendapatkan hasil cluster yang baik, proses

stemming harus di lewati satu kali untuk seluruh dokumen yang akan digunakan, kemudian dokumen hasil *stemming* akan disimpan ke dalam *database*, sehingga dalam proses *clustering* berikutnya proses *preprocessing* ini dapat dilewati dengan menggunakan *dataset* yang telah di-*stemming* sebelumnya.

3.1.2 Clustering

Setelah melakukan *preprocessing*, dilanjutkan dengan tahapan *clustering*, adapun tahapan-tahapan dalam melakukan *clustering* dengan algoritma OC3M yaitu sebagai berikut,

3.1.2.1 Pembentukan Vektor Matriks

Vektor Matriks yang dibentuk terdiri dari dua matriks, yaitu D-Matriks dan C-Matriks. D-Matriks merupakan representasi dari kumpulan kata unik dari semua dokumen. Setiap kolom merepresentasikan kata unik dan setiap baris merupakan dokumen. C-Matriks merupakan relasi tiap dokumen dengan semua dokumen.

3.1.2.2 Penentuan *Seed* dokumen

Dari Vektor Matriks yang terbentuk nilai *decoupling coefficient* dan *coupling coefficient* dapat ditentukan. Berdasarkan nilai dari *decoupling coefficient* dan *decoupling coefficient*, *seed power* dapat dihitung. *Seed power* merupakan nilai acuan untuk menentukan pusat *cluster*. Dari *seed power* semua dokumen, sejumlah dokumen dengan nilai *seed power* tertinggi akan dipilih sebagai pusat *cluster*. Banyaknya dokumen yang akan digunakan sebagai pusat *cluster* berdasarkan jumlah *cluster* yang telah dihitung sebelumnya.

3.1.2.3 Penempatan Dokumen *Nonseed*

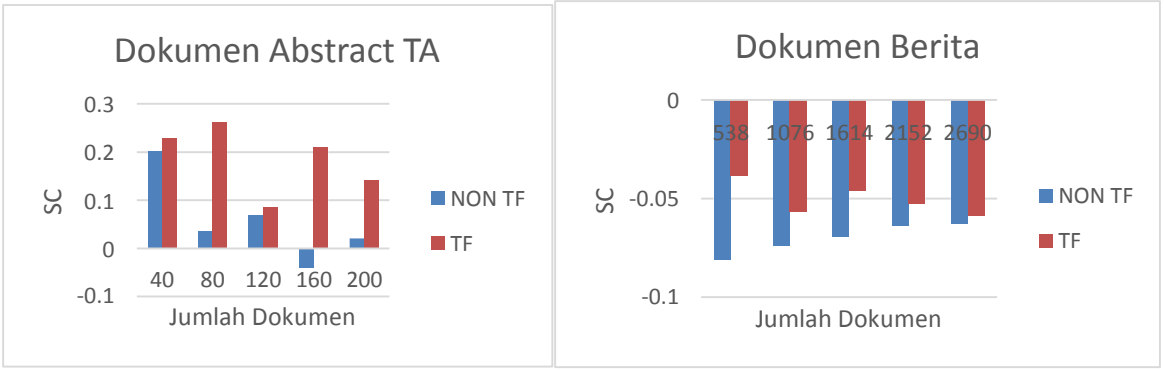
Penempatan dokumen *nonseed* dilakukan dengan membandingkan nilai c_{ij} dokumen *nonseed* dengan nilai c_{ij} dokumen *seed*. Parameter *overlapping coefficient* (*k*) digunakan pada proses ini, parameter ini berguna untuk membatasi suatu dokumen agar dapat melakukan *overlap*. Untuk memastikan terjadinya *overlapping* nilai dari *k* harus lebih besar daripada satu. Dan ketika terjadi *overlapping* digunakan *threshold* untuk mengontrol dokumen yang akan masuk kedalam *cluster* selanjutnya.

3.1.3 Evaluasi *Clustering*

Pada tahapan ini dilakukan suatu pengukuran terhadap kualitas *cluster* yang dihasilkan oleh algoritma OC3M. Adapun metode pengukuran yang digunakan adalah *Silhoutte Coefficient*. Dengan metode ini, akan di ukur nilai *silhouette coefficient* tiap cluster dan nilai rata-rata dari semua *cluster*, dari nilai *silhouette coefficient* ini akan diketahui bagaimana kualitas dari *cluster* berdasarkan pada batasan nilai yang telah di jelaskan sebelumnya

4. Pengujian dan Analisis

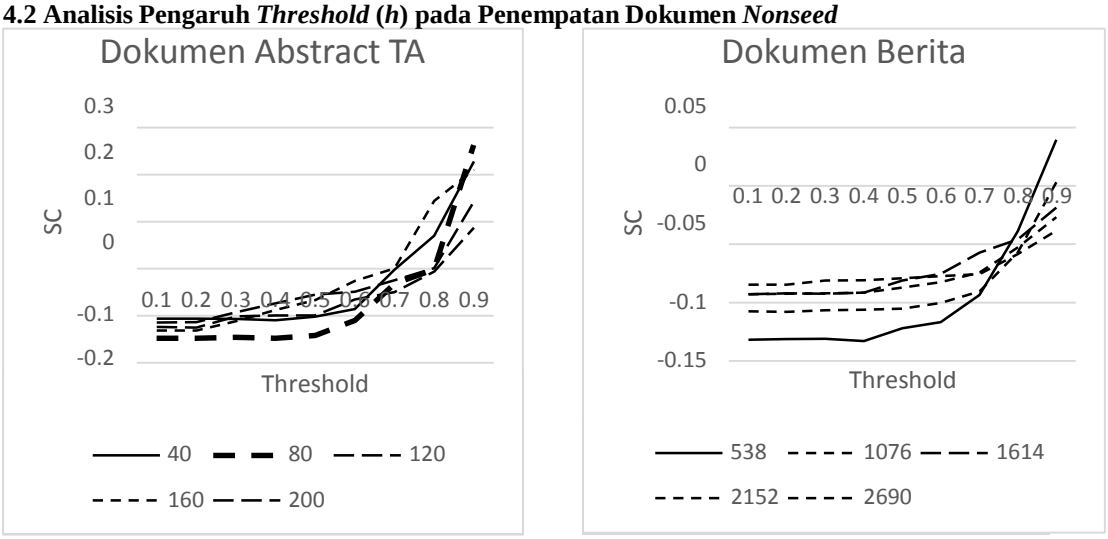
4.1 Analisis Pengaruh *Term Frequency*



Gambar 2. Pengaruh *Term Frequency* pada tiap Dataset

Dari hasil pengujian diatas dapat dilihat bahwa pengaruh dari penggunaan TF terhadap hasil *clustering*, berdasarkan nilai SC *clustering* menggunakan TF lebih bagus dari pada yang tidak menggunakan TF. Ini dapat dilihat pada semua dokumen, dimana ketika menggunakan TF, nilai SC selalu lebih besar dibandingkan jika tidak menggunakan TF.

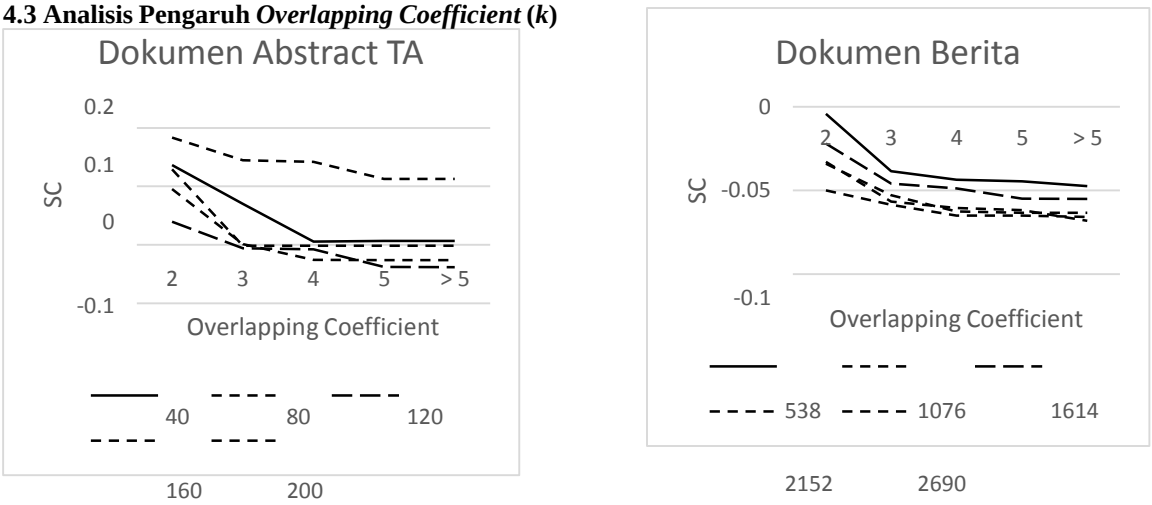
4.2 Analisis Pengaruh *Threshold* (*h*) pada Penempatan Dokumen *Nonseed*



Gambar 3. Pengaruh Parameter *Threshold* (*h*) pada tiap Dataset

Pengujian dilakukan dengan menggunakan nila *overlapping coefficient* = 3. Dari hasil pengujian diatas dapat dilihat bahwa semakin besar nilai *threshold*, maka semakin baik nilai SC yang dihasilkan. Hal ini disebabkan karena dengan menggunakan nilai *threshold* yang tinggi sehingga menyebabkan dokumen yang *overlap* lebih sedikit. Tak hanya itu dengan menggunakan nilai *threshold* yang tinggi menyebabkan dokumen yang *overlap* pun merupakan dokumen-dokumen yang memiliki nilai similaritas yang tinggi dengan pusat *cluster* sehingga nilai SC menjadi lebih baik. Dari grafik diatas dapat dikatakan nilai threshold 0.9 merupakan nilai yang terbaik baik pada dataset berukuran besar maupun kecil.

4.3 Analisis Pengaruh *Overlapping Coefficient* (*k*)



Gambar 4 Pengaruh Parameter *Overlapping Coefficient* (*k*) pada tiap Dataset

Berdasarkan dari hasil pada gambar 4, dapat dikatakan bahwa bertambahnya *overlapping coefficient* akan menyebabkan bertambahnya banyaknya *overlap* yang terjadi sehingga menurunkan nilai SC. Penurunan nilai SC yang signifikan terjadi pada k > 2 dan pada k > 3 rata-rata nilai SC cukup stabil, penurunan yang terjadi sangat sedikit atau tidak ada. Dari grafik diatas, semakin kecil nilai *overlapping coefficient* menyebabkan nilai SC yang semakin bagus, hal ini disebabkan karena dengan semakin kecilnya nilai *overlapping coefficient* maka semakin sedikit pula *overlap* yang terjadi.

4.4 Analisis Kualitas Hasil Cluster

Dari hasil pengujian terhadap parameter-parameter yang telah diuji sebelumnya pada algoritma OC3M, didapat nilai parameter yang optimum pada algoritma ini. Parameter yang optimum dari algoritma ini adalah mengunakan TF,

nilai *threshold* = 0.9, dan nilai *overlapping coefficient* = 2. Dengan menggunakan parameter ini, dianalisis kualitas *cluster* yang dihasilkan.

Kualitas *cluster* yang dihasilkan cukup beragam, kualitas *cluster* yang terbanyak muncul pada *dataset* dokumen Abstrak TA adalah baik, ini dilihat dari banyaknya nilai rata-rata SC yang positif yang dihasilkan. Namun jika dilihat dari rentang nilainya, lebih banyak yang mendekati 0, hal ini disebabkan dengan karena sifat algoritma yang *overlapping* sehingga banyak dokumen yang berada pada batas-batas *cluster*. Jika berdasarkan nilai SC tiap *cluster*-nya, pada hampir pada setiap kelompok *dataset* lebih banyak nilai negatif. Banyaknya nilai negatif ini terjadi karena banyaknya dokumen yang tidak tepat dengan *cluster*-nya, tak hanya itu, sifat *overlap* pada algoritma ini juga mempengaruhi adanya dokumen yang tidak tepat dengan *cluster*-nya. Sedangkan pada *dataset* dokumen berita, nilai rata-rata SC yang dihasilkan cukup baik. Namun seiring dengan bertambahnya data, nilai rata-rata SC semakin memburuk.

5. Kesimpulan

5.1. Kesimpulan

1. Dengan menggunakan *term frequency (tf)* pada pembetukan D-Matriks, maka kualitas cluster menjadi lebih baik namun diperlukan waktu pemrosesan yang lebih lama.
2. Parameter *h* dan *k* sangat mempengaruhi kualitas *cluster* yang terbentuk. Semakin besar nilai *h* yang digunakan maka kualitas *cluster* menjadi lebih baik dan semakin kecil nilai *k* maka kualitas *cluster* menjadi lebih baik.
3. Kualitas *cluster* yang dihasilkan dengan OC3M sangat beragam. Rata-rata nilai SC yang dihasilkan mendekati nilai 0 baik itu postif maupun negatif. Namun jika dilihat antara nilai positif atau negatif, lebih banyak yang bernilai negatif terutama pada nilai SC setiap *cluster*-nya yang dimana nilai negatif berarti kualitas SC untuk setiap *cluster*-nya buruk. Kualitas yang buruk memperlihatkan bahwa terdapat dokumen yang kurang tepat pada *cluster* yang menyelimutinya. Namun jika dirata-ratakan maka hasil *clustering* akan menghasilkan nilai SC yang positif.

4.2 Saran

Dapat menggunakan koleksi dokumen yang ditambahkan secara dinamis sehingga dataset tidak bersifat statis (jumlah data tetap saat proses *clustering*, tanpa ada penambahan data), proses *clustering* dilakukan secara *incremental* sesuai dengan *dataset* yang berubah-ubah.

Daftar Pustaka

[1]. Agusta, Ledy. 2009. Perbandingan Algoritma Stemming Porter dengan Algoritma Nazief & Adriani untuk Stemming Dokumen Teks Bahasa Indonesia. Universitas Kristen Satya Wacana.

[2]. Banerjee, Arindam., Krumpelman, Chase., Ghosh, Joydeep., Basu. Sugato., Mooney, Raymond J. 2005. Model-based Overlapping Clustering. In KDD '05: Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, pages 532–537, New York, NY, USA, 2005. ACM Press.

[3]. Can, Fazli., Altingovde, Ismail Sengor. 2003. Efficiency and Effectiveness of Query Processing in Cluster Based-Retrieval. Bilkent University. Turkey.

[4]. Can, Fazli., Ozkarahan Esen, A. 1990. Concepts and Effectiveness of The Cover Coefficient Based Clustering Methodology for Text Databases. Miami University. Ohio.

[5]. Can, Fazli. 1991. Experiments on Incremental Clustering. Miami University. Ohio.

[6]. Cleuziou, Guillaume. A Generalization of K-Means for Overlapping Clustering. Universite D'Orleans. France

[7]. Rousseeuw, Peter J. 1986. Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. University of Fribourg. Switzerland.

[8]. Ranny, Evalia Roesita. 2010. Analisis dan Implementasi Cluster Based Retrieval Menggunakan Metode Cover Coefficient-Based Clustering Method (C3M) pada Dokumen Teks Berbahasa Inggris. Institut Teknologi Telkom. Bandung.

[9]. Rendon, Erendira., Abundez, Itzel M., Gutierrez, Citlalih., Zagal, Sergio Diaz., Arizmendi, Alejandra., Quiroz, Elvia M., Arzate, Elsa. A Comparison of Internal and External Cluster Validation Indexes. Instituto Tecnologico do Toluca. Mexico.

[10]. Rijsbergen, C. J., 1979, Information Retrieval, Information Retrieval Group, University of Glasgow.

[11]. Rosell, Magnus. 2006. Introduction of Information Retrieval and Text Clustering. Royal Institute. Swedia.

[12]. Zhu, Xinpei. 2003. Automated Hypertext Link Creation Based on Clustered Nodes. Miami University. Ohio.