IMPLEMENTASI METODE IMPROVED FUZZY C-MEANS PADA ANALISIS SENTIMEN REVIEW APLIKASI MOBILE BAHASA INDONESIA

SKRIPSI

Untuk memenuhi sebagian persyaratan untuk mencapai gelar Sarjana Komputer

Disusun oleh:

AGUS SURYANTO

NIM. 115060813111009

KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU KOMPUTER
MALANG
2015

LEMBAR PERSETUJUAN

IMPLEMENTASI METODE IMPROVED FUZZY C-MEANS PADA ANALISIS SENTIMEN REVIEW APLIKASI MOBILE BAHASA INDONESIA

SKRIPSI

Untuk memenuhi sebagian persyaratan untuk mencapai gelar
Sarjana Komputer

Disusun oleh:

AGUS SURYANTO

NIM. 115060813111009

Skripsi ini telah diperiksa dan disetujui oleh:

Pembimbing I	**Pembimbing II**
Indriati, ST., M.Kom	**Drs. Achmad Ridok, M.Kom**
NIK. 831013 06 1 2 0035	**NIP. 19680825 199403 1 004**

repository.ub.ac.id

LEMBAR PENGESAHAN

IMPLEMENTASI METODE IMPROVED FUZZY C-MEANS PADA ANALISIS SENTIMEN REVIEW APLIKASI MOBILE BAHASA INDONESIA

SKRIPSI

Untuk memenuhi sebagian persyaratan untuk mencapai gelar
Sarjana Komputer

Disusun oleh:

AGUS SURYANTO

NIM. 115060813111009

Skripsi ini telah diuji dan dinyatakan lulus pada tanggal 29 Juli 2015

Penguji I	Penguji II
Rekyan Regasari MP, ST., MT.	**Drs. Marji, MT.**
NIK. 770414 06 1 2 0253	**NIP. 19670801 199203 1 001**

Penguji III

M. Ali Fauzi, S.Kom., M.Kom.
NIK. 2015028901011000

Mengetahui,
Ketua Program Studi Informatika / Ilmu Komputer

Drs. Marji, MT.
NIP. 19670801 199203 1 001

UNIVERSITAS BRAWIJAYA

PERNYATAAN
ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepengetahuan Saya di dalam naskah SKRIPSI ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata didalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku. (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70)

Malang, Agustus 2015
Mahasiswa,

Agus Suryanto
NIM. 115060813111009

KATA PENGANTAR

Dengan nama Allah SWT Yang Maha Pengasih dan Penyayang. Segala puji bagi Allah SWT karena atas rahmat dan hidayahNya penulis dapat menyelesaikan tugas akhir yang berjudul "Implementasi Metode Improved Fuzzy C-Means pada Analisis Sentimen Review Aplikasi Mobile Bahasa Indonesia". Shalawat serta salam atas junjungan besar kita Nabi Muhammad S.A.W. beserta keluarga dan para sahabat sekalian.

Tugas akhir ini disusun untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer di Fakultas Ilmu Komputer, Universitas Brawijaya Malang. Melalui kesempatan ini, penulis menyampaikan rasa hormat dan terima kasih penulis yang sebesar-besarnya kepada pihak yang telah memberikan bantuan lahir maupun batin selama penulisan tugas akhir ini. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa hormat dan terima kasih penulis kepada:

1. Kepada orang tua penulis Bapak Saiman dan Ibu Samini dan seluruh keluarga yang senantiasa tiada henti hentinya memberikan kebutuhan jasmani dan rohani demi terselesainya tugas akhir ini.
2. Ibu Indriati, ST., M.Kom. dan Bapak Drs. Achmad Ridok, M.Kom. selaku dosen pembimbing tugas akhir.
3. Bapak Drs. Marji, M.Si. dan Issa Arwani, ST., MT. selaku ketua dan sekretaris program Studi Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya.
4. Seluruh dosen Fakultas Ilmu Komputer Universitas Brawijaya yang telah membekali penulis dengan ilmu-ilmu yang bermanfaat.
5. Seluruh staf Fakultas Ilmu Komputer Universitas Brawijaya.
6. Seluruh teman-teman aliansi Kertosentono 60B: Galih Wicaksono, Alief Maulana, Riyan Wicaksana P., dll.
7. Seluruh teman-teman mahasiswa Fakultas Ilmu Komputer Universitas brawijaya khususnya angkatan 2011.
8. Seluruh pihak yang telah membantu kelancaran penulisan tugas akhir yang tidak dapat penulisan sebutkan satu per satu.

Penulis menyadari bahwa tugas akhir ini masih banyak kekurangan dan masih jauh dari sempurna. Untuk itu, saran dan kritik yang membangun, sangat penulis

harapkan. Semoga tugas akhir ini membawa manfaat bagi penyusun maupun pihak lain yang menggunakannya.

Malang, Juni 2015

Penulis

repository.ub.ac.id

ABSTRAK

Agus Suryanto. 2015. Implementasi Metode Improved Fuzzy C-Means pada Analisis Sentimen *Review* Aplikasi Mobile Bahasa Indonesia. Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya.
Pembimbing: Indriati, ST., M.Kom. dan Drs. Achmad Ridok, M.Kom.

Sentimen analisis adalah ilmu mengenai analisa dari suatu opini, sentimen, evaluasi, penilaian, sikap dan emosi dari sebuah entitas yang dapat berupa produk, layanan, individu, kelompok, isu, peristiwa, dan topik. Analisis sentimen dapat digunakan untuk mengenali sentimen *review* khususnya review aplikasi mobile apakah positif atau negatif. Dari *review* aplikasi terutama *review* negatif, pengembang mampu melakukan perbaikan terhadap aplikasi mereka. Selain itu, *review* juga membantu calon pengguna aplikasi untuk bisa mengetahui seberapa layak suatu aplikasi untuk digunakan, berdasarkan *review-review* yang diberikan oleh pengguna lain yang lebih dulu menggunakan aplikasi tersebut. Sehingga calon pengguna bisa memutuskan apakah akan menggunakan aplikasi tersebut atau tidak. Analisis sentimen dilakukan dalam beberapa tahap yaitu preprocessing (*tokenizing*, normalisasi, *filtering*, dan *stemming*), pembobotan dokumen, dan pengelompokan menggunakan metode Improved Fuzzy C-Means. Dari pengujian variasi jumlah data, nilai *f-measure* tertinggi yang dihasilkan sebesar 86,82% dengan rasio jumlah dokumen yang seimbang antara dokumen positif dengan dokumen negatif. Dan nilai *f-measure* terendah sebesar 78,81% dengan rasio jumlah dokumen yang tidak seimbang. Selain itu, besar kecilnya nilai pembobot tidak berpengaruh terhadap hasil *clustering*.

Kata Kunci: Analisis Sentimen, *Review*, Improved Fuzzy C-Means

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

ABSTRACT

Agus Suryanto. 2015. *Implementation of Improved Fuzzy C-Means of Sentiment Analysis Mobile Aplication Review Indonesian Language. Computer Science, Brawijaya University*.
Advisor: Indriati, ST., M.Kom. *and* Drs. Achmad Ridok, M.Kom.

Sentiment analysis is the science of the analysis of an opinion, sentiment, evaluation, assessment, attitudes and emotions of an entity that can be represented as products, services, individuals, groups, issues, events, and topics. Sentiment analysis can be used to identify sentiment review particularly mobile application review whether positive or negative. From the review of applications particularly negative review, the developer is able to make improvements to their applications. In addition, the review also help potential users of the application to be able to determine how reliable an application to be used, based on the reviews given by other users who had the application. So that potential users can decide whether to use the application or not. Sentiment analysis is performed in several stages, i.e. preprocessing (tokenizing, normalizing, filtering, and stemming), weighting documents, and grouping using Improved Fuzzy C-Means. Based on result of variant data sum, highest f-measure value is 86,82% with balance document ratio between positive document and negative document. And lowest f-measure value is 78,81% with unbalance document ratio. And then value of weight did not affect the result of clustering.

***Keywords**: Sentiment Analisys, Improved Fuzzy C-Means*

UNIVERSITAS BRAWIJAYA

DAFTAR ISI

LEMBAR PERSETUJUAN ii
LEMBAR PENGESAHAN iii
PERNYATAAN ORISINALITAS SKRIPSI iv
KATA PENGANTAR v
ABSTRAK vii
ABSTRACT viii
DAFTAR ISI ix
DAFTAR GAMBAR xii
DAFTAR TABEL xiii
DAFTAR LAMPIRAN xv
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 3
1.3 Batasan Masalah 3
1.4 Tujuan 3
1.5 Manfaat 3
1.6 Sistematika Penulisan 4
BAB II KAJIAN PUSTAKA DAN DASAR TEORI 5
2.1 Kajian Pustaka 5
2.2 Text Mining 6
2.3 Analisis Sentimen 6
2.4 Text Preprocessing 7
2.4.1 Tokenizing 7
2.4.2 Normalisasi 7

2.4.3 Filtering 7
2.4.4 Stemming 8
2.5 Pembobotan 10
2.6 Clustering 11
2.7 Fuzzy C-Means 13
2.8 Improved Fuzzy C-Means 14
2.9 Evaluasi 15
BAB III METODE PENELITIAN DAN PERANCANGAN 17
3.1 Studi Literatur 17
3.2 Pengumpulan Data 18
3.3 Analisis dan Perancangan 18
3.3.1 Kebutuhan Antarmuka 18
3.3.2 Kebutuhan Data 18
3.3.3 Kebutuhan Fungsional 19
3.3.4 Diagram Alir Sistem 19
3.3.5 Desain Antarmuka 28
3.3.6 Manualisasi Analisis Sentimen 29
3.4 Implementasi 53
3.5 Pengujian 53
3.5.1 Skenario Pengujian 53
3.6 Kesimpulan dan Saran 54
BAB IV IMPLEMENTASI 55
4.1 Spesifikasi Sistem 55
4.1.1 Spesifikasi Perangkat Keras 55
4.1.2 Spesifikasi Perangkat Lunak 55
4.2 Batasan Implementasi 55

4.3 Implementasi Algoritma 56
4.3.1 Proses Preprocessing 57
4.3.2 Proses Pembobotan 61
4.3.3 Proses Clustering 63
4.4 Implementasi Antarmuka 68
BAB V PENGUJIAN DAN ANALISIS 73
5.1 Pengujian 73
5.2 Hasil dan Analisis Pengujian 73
5.2.1 Pengujian terhadap Variasi Jumlah Dokumen 73
5.2.2 Pengujian terhadap Variasi Nilai Pembobot 75
5.2.3 Pengujian Metode Bobot Wtf dan Improved Fuzzy C-Means 79
BAB VI KESIMPULAN DAN SARAN 82
6.1 Kesimpulan 82
6.2 Saran 82
DAFTAR PUSTAKA DP-1

DAFTAR GAMBAR

Gambar 3.1 Diagram Alir Penelitian 17
Gambar 3.2 Diagram Alir Keseluruhan 20
Gambar 3.3 Diagram Alir Preprocessing 20
Gambar 3.4 Diagram Alir Tokenizing 21
Gambar 3.5 Diagram Alir Normalisasi 21
Gambar 3.6 Diagram Alir Filtering.....22
Gambar 3.7 Diagram Alir Stemming Arifin-Setiono.....23
Gambar 3.8 Diagram Alir Pembobotan Term Frequency.....24
Gambar 3.9 Diagram Alir Improved Fuzzy C-Means 25
Gambar 3.10 Diagram Alir Pemilihan Pusat Cluster Awal 27
Gambar 3.11 Diagram Alir Penentuan Kelas Cluster 28
Gambar 3.12 Perancangan Antarmuka 28
Gambar 4.1 Interface Input Dokumen 68
Gambar 4.2 Interface Tokenizing 69
Gambar 4.3 Interface Normalisasi 70
Gambar 4.4 Interface Filtering.....70
Gambar 4.5 Interface Stemming 71
Gambar 4.6 Interface Hasil Sentimen Analisis 71
Gambar 4.7 Hasil Proses Improved FCM 72
Gambar 5.1 Grafik Rata-Rata *Precision*, *Recall*, dan *F-measure* Pengujian 1.....75
Gambar 5.2 Grafik Rata-Rata Precision, Recall, dan F-measure Pengujian 2.....78
Gambar 5.3 Grafik F-Measure Bobot Wtf dan Improved Fuzzy C-Means 80

DAFTAR TABEL

Tabel 2.1 Tabel Kontingensi 15
Tabel 3.1 Data Input 29
Tabel 3.2 Tokenizing Data Input 30
Tabel 3.3 Data Input Ternormalisasi 31
Tabel 3.4 Filtering Data Input 32
Tabel 3.5 Stemming Data Input 33
Tabel 3.6 Pembobotan Dokumen D1 33
Tabel 3.7 Pembobotan Dokumen D2 34
Tabel 3.8 Pembobotan Dokumen D3 34
Tabel 3.9 Pembobotan Dokumen D4 34
Tabel 3.10 Pembobotan Dokumen D5 35
Tabel 3.11 Pembobotan Dokumen D6 35
Tabel 3.12 Pembobotan Dokumen D7 36
Tabel 3.13 Pembobotan Dokumen D8 36
Tabel 3.14 Pembobotan Dokumen D9 36
Tabel 3.15 Pembobotan Dokumen D10 36
Tabel 3.16 Pembobotan Dokumen D11 36
Tabel 3.17 Pembobotan Dokumen D12 37
Tabel 3.18 Pembobotan Dokumen D13 37
Tabel 3.19 Pembobotan Dokumen D14 38
Tabel 3.20 Pembobotan Dokumen D15 38
Tabel 3.21 Fitur Dokumen 42
Tabel 3.22 Jarak Antar Dokumen 42
Tabel 3.23 Pemilihan Pusat Cluster Awal Ke-1 43
Tabel 3.24 Jarak Antar Dokumen Yang Tersisa Setelah Mendapatkan C1 43
Tabel 3.25 Pemilihan Pusat Cluster Awal Ke-2 44
Tabel 3.26 Jarak Dokumen Terhadap Pusat Cluster Awal 44
Tabel 3.27 Derajat Keanggotaan Terhadap C1 44
Tabel 3.28 Derajat Keanggotaan Terhadap C2 45
Tabel 3.29 Normalisasi Dan Kuadrat Derajat Keanggotaan 45
Tabel 3.30 Fungsi Obyektif 46

Tabel 3.31 Hasil Clustering.....46
Tabel 3.32 Pusat Cluster Baru.....47
Tabel 3.33 Jarak Dokumen Terhadap Pusat Cluster Baru 48
Tabel 3.34 Derajat Keanggotan Terhadap C1 Baru 48
Tabel 3.35 Derajat Keanggotan Terhadap C2 Baru 49
Tabel 3.36 Normalisasi Dan Kuadrat Derajat Keanggotaan Baru 49
Tabel 3.37 Fungsi Obyektif Baru.....50
Tabel 3.38 Hasil Clustering Baru.....50
Tabel 3.39 Hasil Clustering Akhir 51
Tabel 3.40 Hasil Clustering Akhir 52
Tabel 4.1 Daftar Fungsi Proses Analisis Sentimen.....56
Tabel 5.1 Hasil Pengujian Variasi Jumlah Dokumen 73
Tabel 5.2 Rata-Rata Precision, Recall, F-measure Pengujian 1.....74
Tabel 5.3 Hasil Pengujian Variasi Nilai Pembobot 76
Tabel 5.4 Rata-Rata Precision, Recall, F-measure Pengujian 2.....77
Tabel 5.5 Rata-Rata Precision, Recall, F-measure Pengujian 3.....79

DAFTAR LAMPIRAN

Lampiran 1 Kamus Normalisasi L-1
Lampiran 2 Stopword List L-5
Lampiran 3 Daftar Kata Dasar L-6
Lampiran 4 Daftar Kata Positif L-7
Lampiran 5 Daftar Kata Negatif L-8
Lampiran 6 Pelabelan Data Bagian 1 .....L-9
Lampiran 7 Pelabelan Data Bagian 2 .....L-10
Lampiran 8 Pelabelan Data Bagian 3 .....L-11

BAB I
PENDAHULUAN

1.1 Latar Belakang

Indonesia memiliki 41,3 juta pengguna *smartphone* dan 6 juta pengguna *tablet* pada tahun 2013 [MAU-13]. Dengan jumlah pengguna *smartphone* yang besar, menjadikan Indonesia sebagai pasar yang sangat potensial bagi para pebisnis terutama pengembang aplikasi *mobile*. Android, iOS, dan Windows Phone merupakan tiga *platform smartphone* yang banyak digunakan. Android dan iOS memiliki lebih dari 700 ribu aplikasi di awal tahun 2013. Windows Phone memiliki 150 ribu aplikasi di akhir tahun 2012 [BRA-13].

Setiap toko aplikasi mengijinkan pengguna aplikasi untuk memberikan *review* terhadap aplikasi yang digunakan. *Review* tersebut tidak hanya bermanfaat bagi calon pengguna aplikasi tetapi juga bermanfaat bagi pengembang aplikasi tersebut. Dari sisi pengguna, banyak sedikitnya *review* positif dan negatif dapat mempengaruhi keputusan untuk menggunakan atau tidak menggunakan aplikasi tersebut. Karena banyaknya aplikasi yang ada, akan membingungkan pengguna untuk memilih aplikasi yang terbaik. Dari sisi pengembang, dengan membaca *review* maka pengembang dapat mengetahui fitur apa yang harus diperbaiki atau ditingkatkan.

Penilaian produk aplikasi yang ada saat ini masih bersifat subyektif yaitu menggunakan jumlah bintang yang diberikan oleh *reviewer* itu sendiri. Dan dari toko aplikasi yang ada, tidak ada yang memiliki fitur analisis sentimen untuk menyaring atau mengelompokkan antara *review* positif dan *review* negatif. Analisis sentimen adalah bidang ilmu yang mempelajari bagaimana menganalisa opini, sentimen, evaluasi, penilaian, sikap dan emosi dari sebuah entitas yang dapat berupa produk, pelayanan, organisasi, individu, isu-isu, peristiwa, dan topik [LIU-12]. *Review* dari pengguna aplikasi yang dipengaruhi oleh emosi (sentimen) dikelompokkan atau diklasifikasikan untuk menentukan kepolarisasiannya yaitu positif dan negatif.

Pengelompokan atau pengklasifikasian sentimen dapat dilakukan dengan menggunakan dua teknik yaitu *supervised* dan *unsupervised*. Teknik *supervised* memerlukan data latih, sedangkan *unsupervised* tidak.

Chaovalit dan Zhou telah membandingkan teknik *supervised* dan *unsupervised* pada *review* film. Dengan hasil akurasi 85% untuk *supervised* dan menggunakan pendekatan *machine learning* sebagai metode klasifikasi dan *N-gram* sebagai seleksi fitur. Dan hasil akurasi 77% untuk *unsupervised*, menggunakan pendekatan *Semantic Orientation* (SO) [CHA-05]. Teknik *supervised* memiliki kelemahan yaitu data latih yang digunakan harus berkualitas tinggi untuk setiap tipe dokumen, setiap bidang, dan setiap bahasa. Teknik *unsupervised* lebih cocok digunakan untuk berbagai tipe dokumen dan bidang yang berbeda [NER-12]. Akurasi pada teknik *unsupervised* dapat ditingkatkan dengan melakukan *Improved* (perbaikan) algoritma. Beberapa peneliti telah menerapkan perbaikan algoritma pada teknik *unsupervised* yaitu Fuzzy C-Means (FCM) dan mendapatkan akurasi 88% [DEN-10] dan 89% dengan waktu eksekusi dua kali lebih cepat daripada FCM tradisional [YAN-13].

Jika dibandingkan dengan FCM tradisional, *Improved* FCM memiliki beberapa kelebihan yaitu mampu menemukan jumlah cluster terbaik secara otomatis. Selain itu *Improved* FCM juga mampu menemukan pusat cluster awal mendekati pusat cluster sebenarnya sehingga menjadikan metode ini dua kali lebih cepat daripada FCM tradisional [YAN-13]. Hal tersebut menunjukkan bahwa metode Improved Fuzzy C-Means layak untuk digunakan dalam analisis sentimen.

Analisis sentimen *review* aplikasi dilakukan dalam beberapa tahap. Yang pertama adalah *preprocessing*, yaitu berupa *tokenisasi* (pemisahan kata), *filtering* (pemilihan kata), dan *stemming* (merubah ke kata dasar). Tahap kedua dilakukan pembobotan dokumen. Dan yang terakhir adalah pengelompokan menggunakan metode Improved Fuzzy C-Means.

Berdasarkan uraian tersebut maka perlu dibuat suatu solusi yang akan dikembangkan dalam skripsi ini melalui judul "Implementasi Metode Improved Fuzzy C-Means pada Analisis Sentimen *Review* Aplikasi *Mobile* Bahasa Indonesia" agar *review* aplikasi *mobile* dapat dikelompokkan menjadi *review* positif dan negatif sehingga dapat memberikan keefisienan bagi pengguna dalam memilih aplikasi dan bagi pengembang aplikasi dalam melakukan evaluasi aplikasi.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, rumusan masalah yang akan dikaji dalam penelitian ini adalah sebagai berikut:

1. Mengimplementasikan metode Improved Fuzzy C-Means pada analisis sentimen *review* aplikasi *mobile*.
2. Melakukan pengujian *precision*, *recall*, dan *F-Measure* yang didapatkan dari percobaan analisis sentimen *review* aplikasi mobile menggunakan metode Improved Fuzzy C-Means.

1.3 Batasan Masalah

Batasan-batasan yang digunakan untuk menyelesaikan permasalahan yang sudah dirumuskan sebelumnya antara lain:

1. Obyek yang digunakan adalah 1500 data *review* aplikasi *mobile* dari Google Play Store yaitu "BCA mobile".
2. Data yang digunakan adalah data berbahasa Indonesia.
3. Dokumen yang digunakan sebagai masukan sistem berupa *file* berformat .txt.

1.4 Tujuan

Tujuan yang ingin dicapai dari skripsi ini adalah mengimplementasikan metode Improved Fuzzy C-Means pada sentimen analisis *review* aplikasi *mobile* bahasa Indonesia.

1.5 Manfaat

Manfaat yang bisa diperoleh dari hasil skripsi ini yaitu:

1. Bagi penulis, diharapkan mampu menambah pengalaman di bidang studi terkait dan dapat dijadikan pembelajaran.
2. Bagi ilmu pengetahuan, diharapkan dapat menambah referensi terhadap penelitian baru di bidang studi terkait.
3. Memudahkan pengembang aplikasi untuk mengetahui *review* aplikasi yang telah dikelompokkan. Dengan membaca *review* tersebut terutama *review* negatif, pengembang mampu melakukan perbaikan terhadap aplikasi mereka.
4. Membantu calon pengguna aplikasi untuk bisa mengetahui seberapa layak suatu aplikasi untuk digunakan, berdasarkan *review-review* yang diberikan oleh pengguna lain yang lebih dulu menggunakan aplikasi tersebut. Sehingga

calon pengguna bisa memutuskan apakah akan menggunakan aplikasi tersebut atau tidak.

1.6 Sistematika Penulisan

Sistematika pembahasan dalam penulisan skripsi ini yaitu sebagai berikut:

BAB I Pendahuluan

Berisi tentang latar belakang masalah, perumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan.

BAB II Kajian Pustaka dan Dasar Teori

Berisi tentang teori dasar analisis sentimen, *clustering*, Fuzzy C-Means, Improved Fuzzy C-Means, parameter pengujian *precision*, *recall*, dan *F-measure*.

BAB III Metode Penelitian dan Perancangan

Berisi tentang gambaran umum tahapan-tahapan dalam pengimplementasian Improved Fuzzy C-Means pada analisis sentimen dimulai dari studi literatur, perancangan aplikasi, implementasi aplikasi, serta pengujian dan analisis.

BAB IV Implementasi

Berisi tentang implementasi aplikasi Improved Fuzzy C-Means pada analisis sentimen.

BAB V Pengujian dan Analisis

Berisi tentang proses dan hasil pengujian terhadap sistem yang telah diimplementasikan.

BAB VI Penutup

Berisi tentang kesimpulan dari hasil penelitian, serta saran-saran untuk pengembangan lebih lanjut.

BAB II
KAJIAN PUSTAKA DAN DASAR TEORI

Bab ini terdiri dari dua, yaitu kajian pustaka dan dasar teori. Kajian pustaka merupakan rangkuman singkat mengenai materi terkait yang terdapat di dalam berbagai referensi. Sedangkan dasar teori yang diperlukan pada penelitian ini adalah Sistem Temu Kembali Informasi (STKI), analisis sentimen, *preprocessing* (*tokenization, filtering, stemming*), Improved Fuzzy C-Means, dan evaluasi.

2.1 Kajian Pustaka

Analisis sentimen merupakan bagian dari *text mining*, mengolah data tekstual secara otomatis untuk mendapatkan informasi [PAN-08]. Alec Go dkk. melakukan penelitian pada *microblog* twitter berbahasa Inggris menggunakan *Naive Bayes*, *Maximum Entropy*, dan *SVM* untuk klasifikasi ke dalam dua kelas yaitu positif dan negatif. Dengan menggunakan 1.600.000 data *training* dan 359 data *testing* menghasilkan akurasi di atas 80% menggunakan seleksi fitur *unigram*. Akurasi meningkat 2% dengan menggunakan seleksi fitur *unigram* dan *bigram* sekaligus. Bigram mampu meningkatkan akurasi karena mampu menangani frase yang mengandung kata negasi seperti "not" [GOO-09].

Wicaksono dkk. membandingkan metode *K-Means* dan *Expectation-Maximization* dalam analisis sentimen bahasa Indonesia dengan obyek *review* film Indonesia. Dari 765 dokumen *review*, diperoleh hasil akurasi yang kurang bagus yaitu 71,1% dan 71,5%, dikarenakan pemilihan *genre* film yang kurang spesifik. Sebagai contoh, kata "takut" dalam film dengan *genre horror* merupakan sentimen positif, tetapi menjadi sentimen negatif pada *genre* yang lain [WIC-13].

Penelitian yang berjudul "An Improved Fuzzy Clustering Method for Text Mining" oleh Jiabin Deng dkk. menerapkan perbaikan algoritma pada Fuzzy C-Means (FCM). Dari 500 dokumen berita didapatkan akurasi 88%, meningkat 20% dari FCM tradisional [DEN-10]. Huinjing Yang dkk. juga melakukan perbaikan FCM dan menerapkannya pada 500 data. Didapatkan akurasi sebesar 89% dengan waktu eksekusi dua kali lebih cepat daripada FCM tradisional [YAN-13]. Perbaikan yang dilakukan pada FCM dalam penelitian tersebut adalah dengan memperbaiki proses pemilihan pusat cluster awal. Pusat cluster awal dipilih tidak dengan cara

menentukan nilai acak seperti pada FCM tradisional, melainkan dengan agoritma tersendiri.

Dengan menggunakan metode Improved Fuzzy C-Means, perbedaan skripsi ini dengan penelitian yang dikaji terdapat pada obyek yang digunakan. Obyek yang dipakai pada skripsi ini adalah data *review* aplikasi "BCA mobile" pada Google Play Store. Dengan menggunakan metode pembobotan *term frequency*, data *review* tersebut akan dikelompokkan berdasarkan kepolarisasiannya yaitu positif dan negatif.

2.2 Text Mining

Text mining adalah proses penemuan pola yang sebelumnya tidak terlihat pada dokumen atau sumber tertentu menjadi pola yang diinginkan untuk tujuan tertentu [MUS-09]. *Text mining* sering digunakan untuk analisis informasi, pengambilan keputusan, dan tugas-tugas manajemen informasi lainnya yang terkait dengan data teks dalam jumlah besar. Sistem memanfaatkan peningkatan jumlah data yang tidak terstruktur dalam bentuk teks. Data tersebut diolah untuk memenuhi kebutuhan informasi menggunakan berbagai metode klasifikasi, *clustering*, analisis sentimen, dll.

Text mining adalah bagian dari *data mining*, namun proses *text mining* memerlukan lebih banyak tahapan dibanding *data mining* karena data teks memiliki karakteristik yang lebih kompleks daripada data biasa bahkan data yang sudah terstruktur. Berdasarkan ketidakteraturan struktur data teks, maka proses *text mining* memerlukan beberapa tahap awal yang pada intinya adalah mempersiapkan agar teks dapat diubah menjadi lebih terstruktur.

2.3 Analisis Sentimen

Analisis sentimen adalah bidang ilmu yang mempelajari bagaimana menganalisa opini, sentimen, evaluasi, penilaian, sikap dan emosi dari sebuah entitas yang dapat berupa produk, pelayanan, organisasi, individu, isu-isu, peristiwa, dan topik [LIU-12]. Analisis sentimen biasa digunakan untuk menilai kesukaan atau ketidaksukaan publik terhadap suatu barang atau jasa. Sentimen tersebut merupakan informasi tekstual yang bersifat subjektif dan memiliki polaritas positif dan negatif. Nilai polaritas ini dapat digunakan sebagai parameter dalam menentukan suatu keputusan.

Sebagai contoh, seorang pendatang baru di suatu daerah meminta pendapat orang di daerah tersebut tentang operator selular yang cocok digunakan untuk mengakses internet. Pendapat orang-orang terhadap operator A berbeda-beda, ada yang positif ada yang negatif. Jumlah pendapat positif dan negatif ini akan menentukan keputusan yang akan diambil pendatang baru tersebut dalam memilih operator selular. Jumlah pendapat yang sangat banyak akan membuat pengambilan keputusan menjadi tidak efisien.

2.4 Text Preprocessing

Text preprocessing adalah tahap awal yang digunakan untuk mendapatkan data yang siap untuk diproses. Tahapan dalam *text preprocessing* yaitu: *tokenizing*, *filtering*, dan *stemming*.

2.4.1 Tokenizing

Tokenizing adalah proses pengambilan kata yang menjadi penyusun suatu dokumen. Karakter-karakter pemisah kata akan dihilangkan karena tidak memiliki pengaruh terhadap pemrosesan teks. Karakter-karakter pemisah tersebut bisa berupa spasi, tanda baca, angka, dan karakter selain huruf. Dalam proses ini juga dilakukan pengubahan semua huruf menjadi huruf kecil.

2.4.2 Normalisasi

Normalisasi adalah proses pengubahan kata tidak baku menjadi baku dan kata singkatan menjadi kata aslinya. Hal ini bertujuan untuk memudahkan sistem dalam mengenali banyaknya kata yang bervariasi sehingga sistem menjadi lebih akurat.

2.4.3 Filtering

Filtering adalah proses pemilihan kata dari hasil *tokenizing* yang akan digunakan untuk merepresentasikan suatu dokumen. *Filtering* dapat dilakukan dengan dua cara yaitu *filtering* berdasarkan *stoplist* atau berdasarkan *wordlist*. *Stoplist* adalah kumpulan kata-kata tidak penting. Setiap kata dalam dokumen akan dibandingkan dengan setiap kata dalam *stoplist*. Kata dalam dokumen yang juga merupakan kata dalam *stoplist*, akan dibuang dari hasil *tokenizing*. *Wordlist* adalah kumpulan kata-kata penting yang berpotensi mampu merepresentasikan suatu dokumen. Sistem akan mengambil kata dari dokumen hasil *tokenizing* yang juga merupakan kata dalam wordlist. Dalam pengaplikasiannya, penggunaan stoplist

lebih efisien daripada wordlist. Karena banyaknya kata yang tidak penting jauh lebih sedikit daripada kata yang penting.

2.4.4 Stemming

Proses *stemming* digunakan untuk mengganti bentuk dari suatu kata menjadi kata dasar dan kata tersebut harus sesuai dengan struktur morfologi bahasa Indonesia yang benar [AMI-13]. Penggunaan imbuhan berupa awalan atau akhiran pada suatu kata akan dihapus sehingga akan mendapatkan inti dari kata tersebut. Salah satu algoritma *stemming* yaitu *stemming* Arifin-Setiono. Algoritma ini didahului dengan pembacaan tiap kata dari file sampel, sehingga input dari algoritma ini adalah sebuah kata, yang kemudian dilakukan [ARI-02]:

1. Pemeriksaan semua kemungkinan bentuk kata. Setiap kata diasumsikan memiliki dua awalan (prefiks) dan tiga akhiran (sufiks), sehingga bentuknya menjadi:

   Prefiks 1 + prefiks 2 + kata dasar + sufiks 3 + sufiks 2 + sufiks 1

   Seandainya kata tersebut tidak memiliki imbuhan sebanyak imbuhan di atas, maka imbuhan yang kosong diberi tanda x untuk prefiks dan diberi tanda xx untuk sufiks.
2. Pemotongan dilakukan secara berurutan sebagai berikut:

   AW : awalan.

   AK : akhiran.

   KD : kata dasar.

   AW I, hasilnya disimpan pada p1,

   AW II, hasilnya disimpan pada p2,

   AK I, hasilnya disimpan pada s1,

   AK II, hasilnya disimpan pada s2,

   AK III, hasilnya disimpan pada s3.

   Pada setiap tahap pemotongan di atas diikuti dengan pemeriksaan di kamus apakah hasil pemotongan itu sudah berada dalam bentuk dasar. Kalau pemeriksaan ini berhasil maka proses dinyatakan selesai dan tidak perlu melanjutkan proses pemotongan imbuhan lainnya. Contoh pemenggalan kata "mempermainkannya":

   Langkah 1:

Cek apakah kata ada dalam kamus.

Ya: sukses.

Tidak: lakukan pemotongan AW I.

Kata = permainkannya.

Langkah 2:

Cek apakah kata ada dalam kamus.

Ya: sukses.

Tidak: lakukan pemotongan AW II.

Kata = mainkannya.

Langkah 3:

Cek apakah kata ada dalam kamus.

Ya: sukses.

Tidak: lakukan pemotongan AK I.

Kata = mainkan.

Langkah 4:

Cek apakah kata ada dalam kamus.

Ya: sukses.

Tidak: lakukan pemotongan AK II.

Kata = main.

Langkah 5:

Cek apakah kata ada dalam kamus.

Ya: sukses.

Tidak: lakukan pemotongan AK III. Dalam hal ini AK III tidak ada, sehingga kata tidak diubah.

Kata = main.

Langkah 6:

Cek apakah kata ada dalam kamus.

Ya: sukses.

Tidak: “Kata tidak ditemukan”

3. Namun jika sampai pada pemotongan AK III belum juga ditemukan di kamus, maka dilakukan proses kombinasi. KD yang dihasilkan dikombinasikan dengan imbuhan-imbuhan dalam 12 konfigurasi berikut:

a. KD
b. KD + AK III
c. KD + AK III + AK II
d. KD + AK III + AK II + AK I
e. AW I + AW II + KD
f. AW I + AW II + KD + AK III
g. AW I + AW II + KD + AK III + AK II
h. AW I + AW II + KD + AK III + AK II +AK I
i. AW II + KD
j. AW II + KD + AK III
k. AW II + KD + AK III + AK II
l. AW II + KD + AK III + AK II + AK I

Sebenarnya kombinasi a, b, c, d, h, dan l sudah diperiksa pada tahap sebelumnya, karena kombinasi ini adalah hasil pemotongan bertahap tersebut. Dengan demikian, kombinasi yang masih perlu dilakukan tinggal enam yaitu pada kombinasi e, f, g, i, j, dan k. Tentunya bila hasil pemeriksaan suatu kombinasi telah ditemukan kata dasar, maka pemeriksaan pada kombinasi lainnya sudah tidak diperlukan lagi.

2.5 Pembobotan

Pembobotan adalah proses merubah *term* yang merupakan data kualitatif menjadi data kuantitatif sehingga bisa diproses oleh komputer. Pembobotan dilakukan pada dokumen yang sudah dilakukan *preprocessing*. Beberapa metode pembobotan yang umum digunakan yaitu bembobotan TF dan TF-IDF.

Term frequency (TF) adalah jumlah kemunculan kata pada suatu dokumen. Semakin banyak suatu kata muncul pada dokumen, maka semakin besar kata tersebut berpengaruh pada dokumen tersebut. Sebaliknya, semakin sedikit suatu kata muncul pada dokumen, maka semakin kecil kata tersebut berpengaruh pada dokumen tersebut. Pembobotan menggunakan TF dijelaskan pada Persamaan (2-1) [FEL-07].

$$w_{tf(t,d)} = 1 + \log TF(t,d) \quad (2\text{-}1)$$

Keterangan:

$w_{tf(t,d)}$ = bobot *term* t pada dokumen d,

TF(t,d) = jumlah kemunculan kata t pada dokumen d.

Inverse document frequency (IDF) adalah pembobotan kata yang didasarkan pada banyaknya dokumen yang mengandung kata tertentu. Semakin banyak dokumen yang mengandung suatu kata tertentu, semakin kecil pengaruh kata tersebut pada dokumen. Sebaliknya, semakin sedikit dokumen yang mengandung suatu kata tertentu, semakin besar pengaruh kata tersebut pada dokumen. Pembobotan menggunakan TF-IDF dijelaskan pada Persamaan (2-2) [FEL-07].

$$w(t,d) = TF(t,d) * log\frac{D}{Dt} \quad (2\text{-}2)$$

Keterangan:

w(t,d) = bobot kata t pada dokumen d,

TF(t,d) = jumlah kemunculan kata t pada dokumen d,

D = jumlah seluruh dokumen,

Dt = Jumlah dokumen yang mengandung kata t.

2.6 Clustering

Clustering adalah suatu metode pengelompokan berdasarkan ukuran kedekatan (kemiripan). *Cluster* merupakan pola yang terbentuk dalam suatu proses pembagian sekelompok data ke dalam sejumlah sub-kelas. *Clustering* dapat digunakan untuk memberikan label suatu data yang belum diketahui kelasnya. Oleh karena itu, clustering digolongkan sebagai *unsupervised learning* [HAN-20]. *Clustering* bekerja dengan cara memaksimalkan kesamaan antar anggota dalam suatu kelompok dan meminimalkan jarak antara pusat *cluster* dengan setiap anggota dalam *cluster* tersebut.

Clustering dapat diterapkan ke dalam data yang kuantitatif (numerik), kualitatif (kategorikal), atau kombinasi dari keduanya. Dalam terminologi pengenalan pola, kolom dalam matriks menunjukkan *patterns* atau *objects*, baris menunjukkan *feature* atau atribut, dan Z disebut data matriks yang direpresentasikan sebagai matriks n x N dengan N adalah banyaknya object dan n adalah banyaknya atribut.[KUS-09].

$$Z = \begin{bmatrix} z_{11} & \cdots & z_{1N} \\ \vdots & \ddots & \vdots \\ z_{n1} & \cdots & z_{nN} \end{bmatrix}$$

Cluster merupakan wujud himpunan bagian dari suatu himpunan data, dan metode *clustering* dapat diklasifikasikan menjadi dua, yaitu *hard (crisp)* dan *fuzzy*. Metode *hard clustering* merupakan model yang berdasar pada teori himpunan klasik, yang mana suatu obyek menjadi anggota atau tidak menjadi anggota secara penuh ke dalam suatu kelompok. Sebaliknya, metode *fuzzy clustering* mengijinkan obyek untuk menjadi bagian dari beberapa kelompok secara bersamaan dengan perbedaan derajat keanggotaan.

Sebagai contoh dalam *hard clustering*, himpunan data Z = {z_1, z_2, ..., z_{10}} jika dibagi menjadi dua kelompok, maka himpunan U yang merupakan matriks partisi yang menunjukkan derajat keanggotaan elemen himpunan Z dalam kelompok A_1 atau A_2 akan berwujud sebagai berikut [KUS-09]:

$$U = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Baris atas matriks partisi U menunjukkan derajar keanggotaan elemen himpunan Z dalam A_1, dan baris bawah menunjukkan derajat keanggotaan elemen Z dalam A_2. Tampak bahwa setiap elemen himpunan Z secara khusus atau penuh akan menjadi salah satu anggota kelompok (A_1 atau A_2) dengan derajat keanggotaan 1 dan tidak menjadi anggota dalam kelompok yang lain dengan derajat keanggotaan 0. x_1, x_2, x_3, x_4, x_5, dan x_6 secara khusus merupakan anggota dari A_1, sedangkan x_7, x_8, x_9, dan x_{10} secara khusus merupakan anggota dari kelompok A_2.

Sementara di dalam *fuzzy clustering*, derajat keanggotaan data dalam suatu kelompok bukan hanya 0 dan 1, tetapi dapat memiliki nilai antara 0 hingga 1. Baris ke-i dalam matriks partisi menunjukkan derajat keanggotaan data terhadap kelompok Ai. Nilai derajat keanggotaan dalam setiap kolom matriks partisi selalu berjumlah 1 untuk setiap data.

Sebagai contoh, himpunan Z = {z_1, z_2, ..., z_{10}} yang dibagi menjadi dua kelompok Z_1 dan Z_2, maka matriks partisi U dapat dituliskan seperti berikut:

$$U = \begin{bmatrix} 1.0 & 1.0 & 1.0 & 1.0 & 0.7 & 0.4 & 0.2 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.3 & 0.6 & 0.8 & 1.0 & 1.0 & 1.0 \end{bmatrix}$$

Baris atas matriks partisi U menunjukkan derajat keanggotaan elemen himpunan Z dalam A_1 dan baris bawah menjukkan derajat keanggotaan elemen himpunan Z dalam A_2.

2.7 Fuzzy C-Means

Fuzzy C-Means (FCM) merupakan salah satu teknik pengelompokan yang mengijinkan setiap data menjadi anggota dari dua atau lebih cluster. Metode ini dikembangkan oleh Dunn pada tahun 1973 dan diperbaiki oleh Bezdek pada tahun 1981 [WAN-10]. Konsep dasar FCM, pertama kali adalah menentukan pusat *cluster* yang ditentukan secara acak dan tidak akurat. Tiap-tiap data memiliki derajat keanggotaan terhadap semua *cluster*. Pusat *cluster* dan derajat keanggotaan akan diperbaiki secara berulang sedemikian hingga menemukan pusat *cluster* yang tepat dan derajat keanggotaan tiap-tiap data mencapai nilai yang konvergen atau tidak berubah-ubah. Algoritma dari FCM adalah sebagai berikut [YAN-13]:

1. Tentukan bilangan random sebagai pusat *cluster* awal (V_i) dengan i = 1, 2, 3, ..., c. Dengan c adalah banyaknya *cluster*.
2. Tentukan: iterasi awal (t), jumlah *cluster* (c), pembobot (m), maksimum iterasi, toleransi penghentian (ε), fungsi obyektif awal (J_0).
3. Hitung nilai derajat keanggotaan ($u_{i,k}$) pada data ke-i terhadap pusat *cluster* ke-k, dengan menggunakan Persamaan (2-3).

$$u_{i,k}(t) = \frac{1}{\sum_{k=1}^{c}\left(d_{i.j}(t)/d_{k,j}(t)\right)^{2/(m-1)}} \quad (2\text{-}3)$$

   Dengan $d_{i,j}(t)$ adalah jarak data ke-i terhadap pusat *cluster* ke-j pada iterasi ke-t, dan $d_{k,j}(t)$ adalah jarak pusat *cluster* ke-k terhadap pusat *cluster* ke-j pada iterasi ke t. Jika $d_{i,j}(t)=0$, maka $u_{ij}=1$ dan $u_{r,j}=0$ dengan ($r \neq j$). Jarak antar data dihitung menggunakan Euclidean distance seperti pada Persamaan (2-4) [DEN-10].

$$d_{i.j} = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \cdots + (x_{ip} - x_{jp})^2} \quad (2\text{-}4)$$

   Dengan x adalah nilai atribut ke-p pada data ke-i atau ke-j, di mana p adalah banyaknya atribut.

4. Hitung pusat *cluster* (V_i) pada iterasi t+1 menggunakan Persamaan (2-5).

$$V_i(t+1) = \frac{\sum_{j=1}^{n}\left({u_{i,j}}^m(t) * x_{j,p}\right)}{\sum_{j=1}^{n}{u_{i,j}}^m(t)} \quad (2\text{-}5)$$

   Dengan n adalah banyaknya data.

5. Hitung kembali derajat keanggotaan menggunakan Persamaan (2-3).

6. Hitung fungsi obyektif pada iterasi ke t (J) berdasarkan Persamaan (2-6).

$$J(t) = \sum_{i=1}^{n}\sum_{j=1}^{c}\left(u_{i,j}^{m} * {d_{i,j}}^{2}\right) \quad (2\text{-}6)$$

7. Cek kondisi:
   - Jika $|J(t) - J(t\text{-}1)| < \varepsilon$) atau t > maksimum iterasi, maka berhenti;
   - Jika tidak memenuhi kondisi di atas, t = t+1, dan kembali ke langkah 4.

Fuzzy C-Means (FCM) memiliki kelebihan yaitu mampu mengelompokkan lebih dari satu obyek secara bersamaan dan tanpa memerlukan data latih. Selain itu, FCM mampu menghasilkan keluaran yang konvergen. Tetapi FCM juga memiliki kekurangan yaitu waktu untuk mencapai hasil yang konvergen yang lama ketika salah memilih pusat cluster awal, dikarenakan penentuannya yang dilakukan secara acak. Selain itu, FCM rentan terhadap data outlier yaitu data yang jauh dari persebaran data lainnya [SHA-12]. Untuk menutupi kekurangan tersebut, maka digunakan Improved Fuzzy C-Means.

2.8 Improved Fuzzy C-Means

Improved Fuzzy C-Means menggunakan metode tersendiri untuk menemukan lokasi pusat *cluster* awal beserta jumlah cluster terbaik. Jika pusat *cluster* awal yang dipilih dekat dengan pusat *cluster* yang sebenarnya, maka kecepatan untuk mendapatkan hasil yang konvergen dan keakurasian dari algoritma dapat ditingkatkan. Secara umum dapat dikatakan bahwa, pusat *cluster* dapat ditemukan pada area yang memiliki persebaran data yang besar. Sehingga, untuk menemukan pusat cluster awal, harus memenuhi dua kondisi sebagai berikut:

1. *Cluster* terbaik adalah *cluster* yang memiliki kepadatan data tertinggi,
2. Semakin jauh jarak antara pusat *cluster* yang satu dengan yang lainnya, semaik baik.

Untuk mendapatkan pusat *clester* awal, yaitu sebagai berikut [YAN-13]:

1. Tentukan nilai *threshold* (k=3). Nilai *threshold* menunjukkan banyaknya obyek minimal sehingga bisa ditentukan salah satu dari obyek tersebut untuk menjadi pusat *cluster*.
2. Hitung matriks jarak (D) menggunakan persamaan (2-4):

$$D = \begin{bmatrix} d_{1,1} & \cdots & d_{1,n} \\ \vdots & \ddots & \vdots \\ d_{n,1} & \cdots & d_{n,n} \end{bmatrix}$$

3. Tentukan jarak rata-rata ($D'i$) dari masing-masing obyek ke obyek yang lain menggunakan Persamaan (2-7). n adalah banyaknya obyek.

$$D'i = \frac{\sum_{k=1}^{n} d_{i,k}}{n-1} \quad (2\text{-}7)$$

4. Hitung kepadatan (Pi) dari obyek x_i menggunakan Persamaan (2-8).

$$P_i = \frac{t_i}{D_{i-max}} \quad (2\text{-}8)$$

Dengan ti adalah banyaknya obyek yang jaraknya kurang dari $D'i$ dan Di-max adalah jarak terbesar dari sekian obyek yang jaraknya kurang dari $D'i$. Semakin besar nilai Pi, semakin banyak obyek yang berada di sekitar obyek xi. Dan sebaliknya, semakin kecil nilai Pi, semakin sedikit obyek yang berada di sekitar obyek xi.

5. Pilih pusat *cluster* pertama dari obyek yang memiliki nilai Pi terbesar.
6. Hapus obyek yang menjadi pusat *cluster* dan obyek yang memiliki jarak dengan pusat *cluster* kurang dari atau sama dengan Di-max dari martiks D.
7. Hitung nilai n baru yaitu n-t_i dari t_i dokumen terpilih. Jika n<k, ulangi langkah di atas, selain itu proses berhenti dan didapat pusat *cluster* awal sebanyak jumlah *cluster* terbaik. Pusat *cluster* yang telah didapatkan tersebut akan menjadi *input* untuk diproses menggunakan algoritma FCM.

2.9 Evaluasi

Evaluasi adalah kegiatan mengukur keberhasilan hasil implementasi dengan membandingkan kinerja sistem dengan hasil yang diinginkan (biasanya dari pakar), menggunakan metode tertentu dalam kondisi yang berbeda-beda. Dalam sistem temu kembali informasi, standar yang digunakan dalam mengukur kinerja sistem yaitu *precision* dan *recall*.

Nilai *precision* dan *recall* bisa didapatkan dari perbandingan tabel kontingensi. Tabel kontingensi adalah tabel yang digunakan untuk mengukur hubungan antara dua variabel seperti pada Tabel 2.1.

Tabel 2.1 Tabel Kontingensi

	Actual Class	
	+	-

Predicted Class	+	TP	FP
	-	FN	TN

Sumber: [MAN-09]

Keterangan:

- TP (True Positive) menunjukkan banyaknya dokumen yang berdasarkan hasil klasifikasi sistem termasuk ke dalam suatu kelas, dan berdasarkan hasil klasifikasi manual memang merupakan anggota kelas tersebut.
- FN (False Negative) menunjukkan banyaknya dokumen yang berdasarkan hasil klasifikasi manual termasuk ke dalam suatu kelas, tetapi berdasarkan hasil klasifikasi sistem bukan merupakan anggota kelas tersebut.
- FP (False Positif) menunjukkan banyaknya dokumen yang berdasarkan hasil klasifikasi sistem bukan termasuk ke dalam suatu kelas, tetapi berdasarkan hasil klasifikasi manual merupakan anggota kelas tersebut.
- TN (False Positif) menunjukkan banyaknya dokumen yang berdasarkan hasil klasifikasi sistem bukan termasuk ke dalam suatu kelas, dan berdasarkan hasil klasifikasi manual memang bukan merupakan anggota kelas tersebut.

Precision adalah tingkat kesesuaian antara informasi yang diminta oleh pengguna dengan hasil yang diberikan oleh sistem. Persamaan (2-9) merupakan persamaan dari *precision*.

$$Precision = \frac{TP}{TP + FP} \quad (2\text{-}9)$$

Recall adalah tingkat kesesuaian antara informasi yang diperoleh dari sistem dengan informasi yang diinginkan pengguna. Persamaan (2-10) merupakan persamaan dari *recall*.

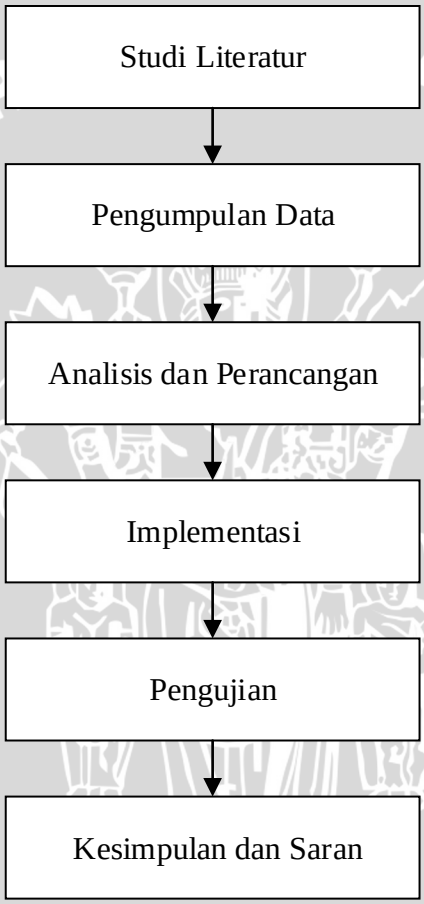
$$Recall = \frac{TP}{TP + FN} \quad (2\text{-}10)$$

F-Measure adalah pengukuran untuk mengetahui tingkat keseimbangan antara *presicion* dan *recall*. Perhitungan *F-Measure* ditunjukkan pada Persamaan (2-11).

$$F - Measure = \frac{2\ x\ Precision\ x\ Recall}{Precision + Recall} \quad (2\text{-}11)$$

BAB III
METODE PENELITIAN DAN PERANCANGAN

Bab ini berisi tentang langkah-langkah yang dilakukan dalam penelitian yang berjudul "Implementasi Metode Improved Fuzzy C-Means pada Analisis Sentimen *Review* Aplikasi *Mobile* Bahasa Indonesia". Langkah-langkah tersebut antara lain: studi literature, pengumpulan data, analisis dan perancangan, implementasi, dan pengujian. Secara umum tahapan dalam penelitian ini ditunjukkan pada Gambar 3.1.



Gambar 3.1 Diagram Alir Penelitian

3.1 Studi Literatur

Studi literatur bertujuan untuk mempelajari mengenai penelitian yang mendukung dan teori yang mendasari implementasi Improved Fuzzy C-Means untuk menyelesaikan permasalahan analisis sentimen *review* aplikasi *mobile* bahasa Indonesia. Teori-teori pendukung tersebut diperoleh dari buku, jurnal, skripsi terkait. Teori-teori tersebut antara lain membahas tentang:

1. *Text mining*,
2. Analisis sentimen,
3. Metode Fuzzy C-Means dan perbaikannya, dan
4. Pengembangan aplikasi Android.

3.2 Pengumpulan Data

Penelitian ini menggunakan data yang didapatkan dari Google Play Store, yang merupakan toko aplikasi dari sistem operasi perangkat mobile Android. Data tersebut merupakan *review* dari 1500 pengguna aplikasi "BCA Mobile". Untuk menguji keefektifan sistem, hasil *clustering* dari data tersebut akan dibandingkan dengan data aktual. Pelabelan data aktual dilakukan dengan cara memberikan kuisioner kepada tiga orang untuk setiap *review*. Hasil pelabelan didapatkan dengan mengambil suara terbanyak dari ketiga responden tersebut. Contoh pelabelan ditunjukkan pada Lampiran 6 hingga Lampiran 8.

3.3 Analisis dan Perancangan

Analisis kebutuhan bertujuan untuk mendapatkan semua kebutuhan yang diperlukan oleh perangkat lunak yang akan dibangun sehingga mampu menyelesaikan permasalahan dengan baik. Analisis kebutuhan dilakukan dengan mengidentifikasi kebutuhan sistem dan apa saja yang terlibat di dalamnya. Analisis kebutuhan dalam sistem ini antara lain:

3.3.1 Kebutuhan Antarmuka

Kebutuhan antarmuka dalam pengembangan perangkat lunak yang akan dibangun meliputi:

1. Perangkat lunak memiliki tampilan yang mudah dimengerti oleh pengguna (*user friendly*).
2. Perangkat lunak mampu menampilkan *review* yang belum dikelompokkan.
3. Perangkat lunak mampu menampilkan hasil analisis sentimen yang telah diproses

3.3.2 Kebutuhan Data

Data yang diolah oleh perangkat lunak ini adalah:

1. Data *review* salah satu aplikasi *mobile* yaitu "BCA Mobile" yang didapatkan dari toko aplikasi Google Play Store sebanyak 500 *review*.

2. Data kata baku yang dibutuhkan untuk pengubahan kata tidak baku dalam dokumen *review* menjadi kata baku yang dihimpun berdasarkan Kamus Besar Bahasa Indonesia (KBBI).
3. Data kata dasar bahasa Indonesia yang dibutuhkan dalam proses *stemming* yang dihimpun berdasarkan Kamus Besar Bahasa Indonesia (KBBI).
4. Data *stop word* bahasa Indonesia yang dibutuhkan dalam proses *filtering* yang dihimpun secara manual.

3.3.3 Kebutuhan Fungsional

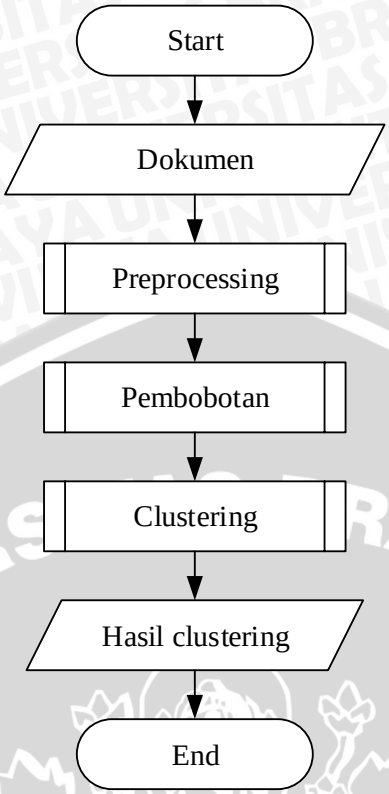
Perangkat lunak ini akan mampu menjalankan fungsi-fungsi sebagai berikut:

1. Mampu melakukan proses preprocessing,
2. Mampu melakukan proses pembobotan setiap dokumen, dan
3. Mampu mengelompokkan dokumen menggunakan Improved Fuzzy C-Means.

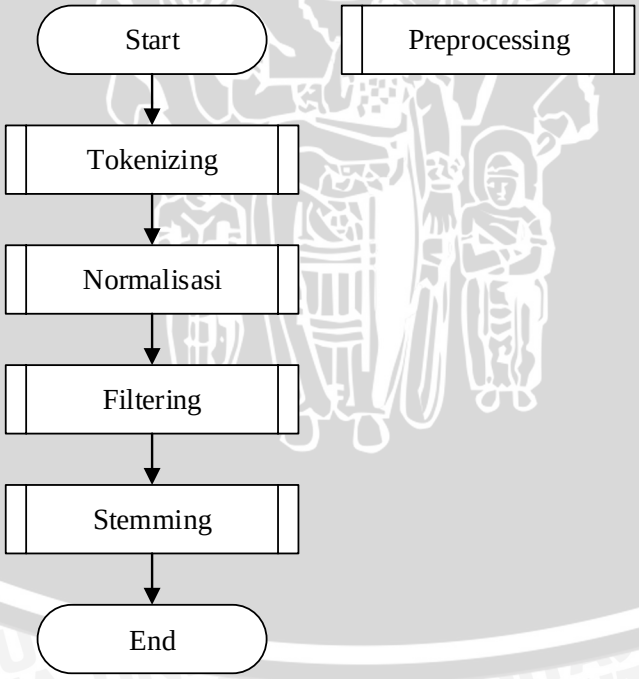
3.3.4 Diagram Alir Sistem

Diagram alir sistem menunjukkan proses-proses jalannya aplikasi. Diagram alir aplikasi analisis sentimen *review* aplikasi *mobile* bahasa Indonesia menggunakan metode Improved Fuzzy C-Means ditunjukkan pada Gambar 3.2.

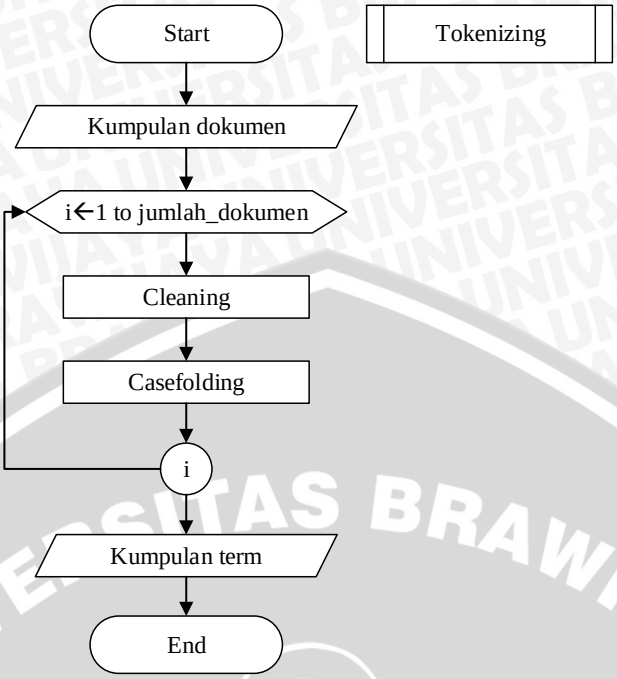
Pertama yang dibutuhkan oleh sistem adalah dokumen *review* berupa sebuah file berformat .txt. Isi dari file berupa beberapa baris *review*. Satu baris *review* merepresentasikan satu dokumen *review*. Kemudian, dokumen tersebut akan dilakukan *preprocessing*. Tahap-tahap *preprocessing* terdiri dari *tokenizing*, normalisasi, *filtering*, dan *stemming* yang digambarkan dalam diagram alir seperti pada Gambar 3.3. Setelah *preprocessing* selesai, dokumen akan dilakukan pembobotan menggunakan metode pembobotan *term frequency* (W_{tf}) sehingga bisa dilakukan proses *clustering*. *Clustering* dilakukan menggunakan metode Improved Fuzzy C-Means untuk menentukan polaritas sentimen. Hasil dari *clustering* berupa kelompok *review* positif dan negatif.



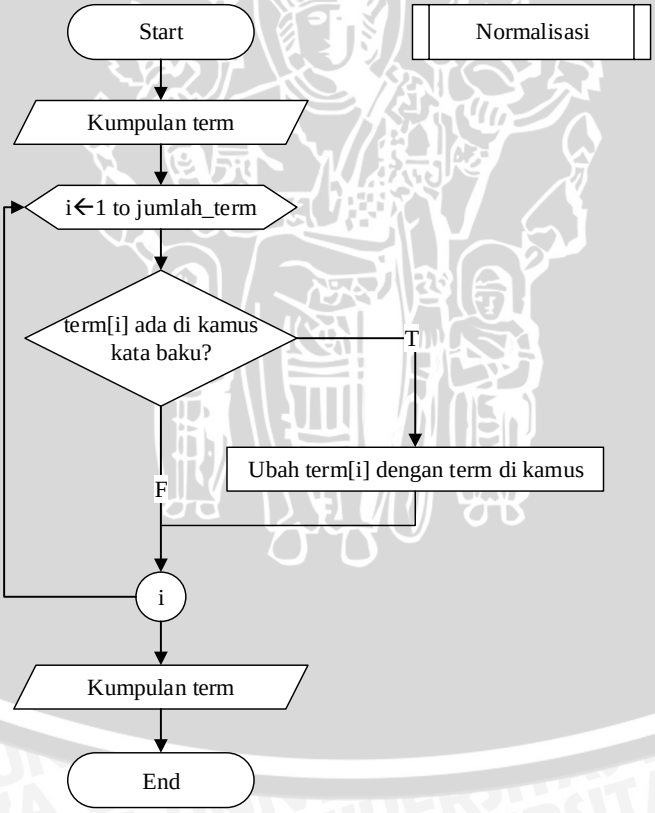
Gambar 3.2 Diagram Alir Keseluruhan



Gambar 3.3 Diagram Alir Preprocessing



Gambar 3.4 Diagram Alir Tokenizing

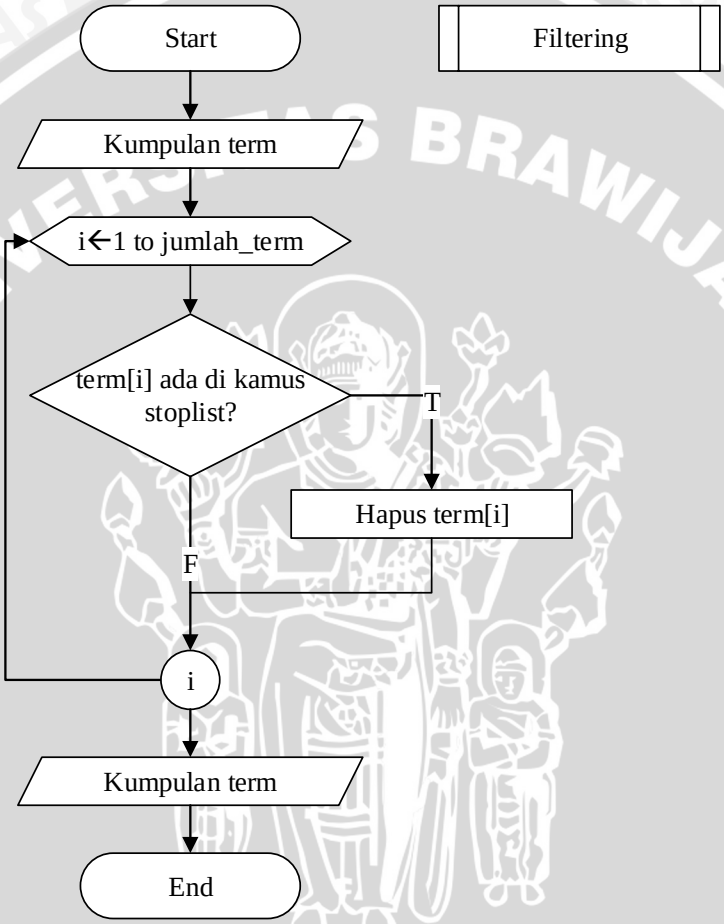


Gambar 3.5 Diagram Alir Normalisasi

Tokenizing terdiri dari *cleaning* dan *casefolding*. *Cleaning* bertujuan untuk menghilangkan karakter-karakter selain huruf alfabet seperti angka dan tanda baca, karena karakter tersebut tidak memiliki hubungan dengan informasi yang ada pada

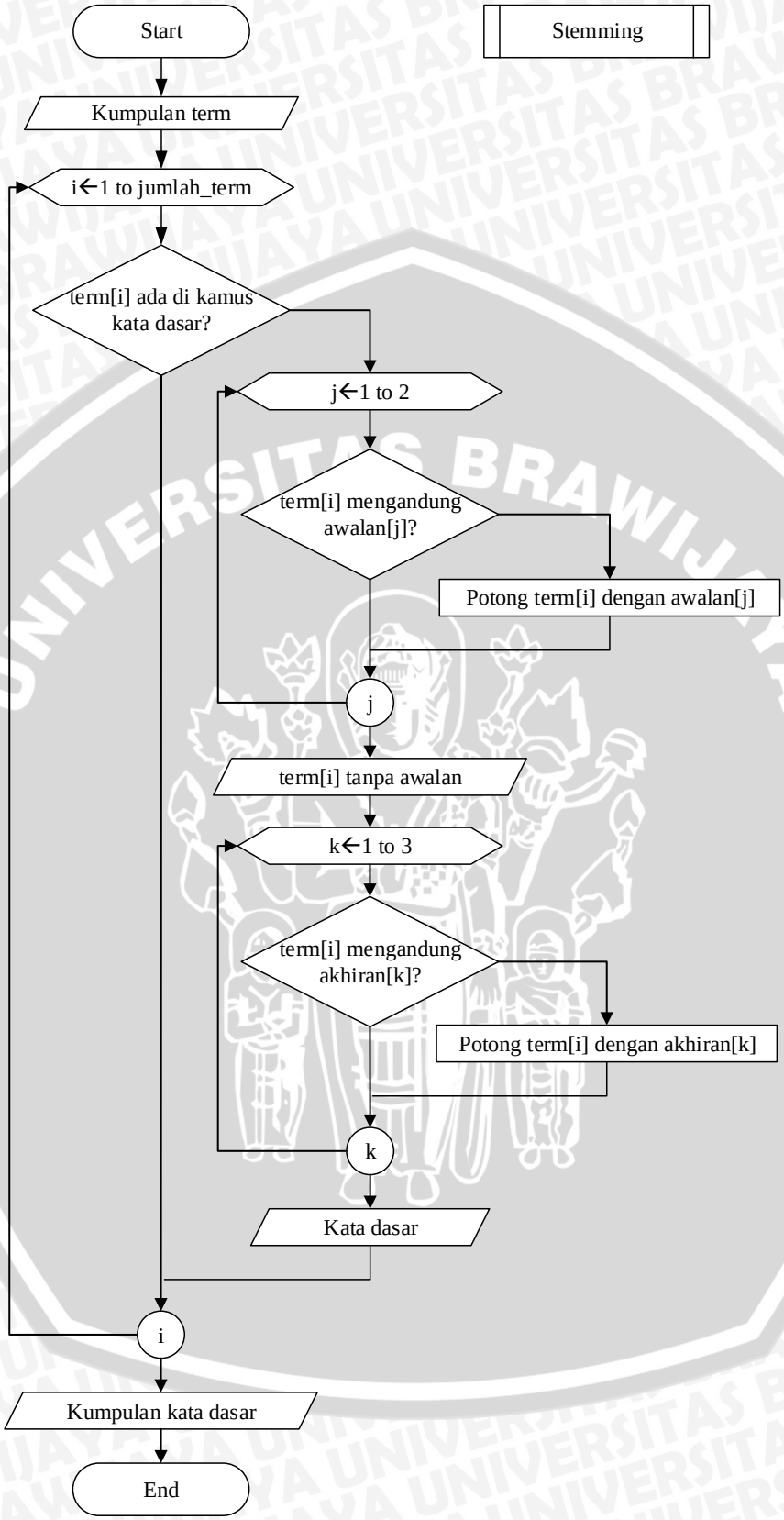
dokumen. Sedangkan *casefolding* bertujuan untuk mengubah semua karakter menjadi huruf kecil. Diagram alir *tokenizing* ditunjukkan pada Gambar 3.4.

Setelah dokumen dilakukan *tokenizing*, proses selanjutnya adalah normalisasi. Normalisasi bertujuan untuk mengubah semua kata tidak baku menjadi kata baku. Pengubahan kata tersebut ditentukan berdasarkan kamus kata baku. Diagram alir dari normalisasi ditunjukkan seperti pada Gambar 3.5.

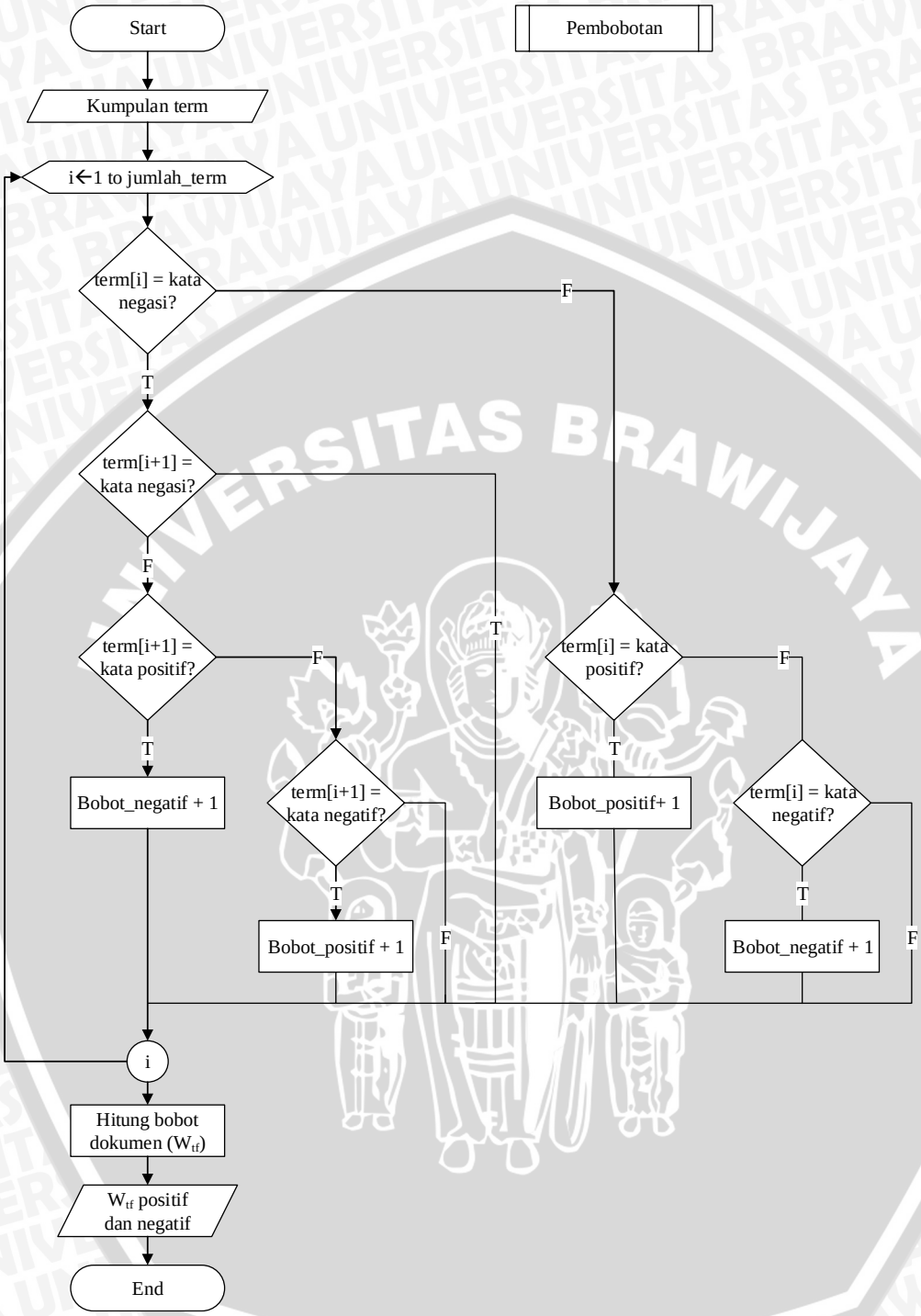


Gambar 3.6 Diagram Alir Filtering

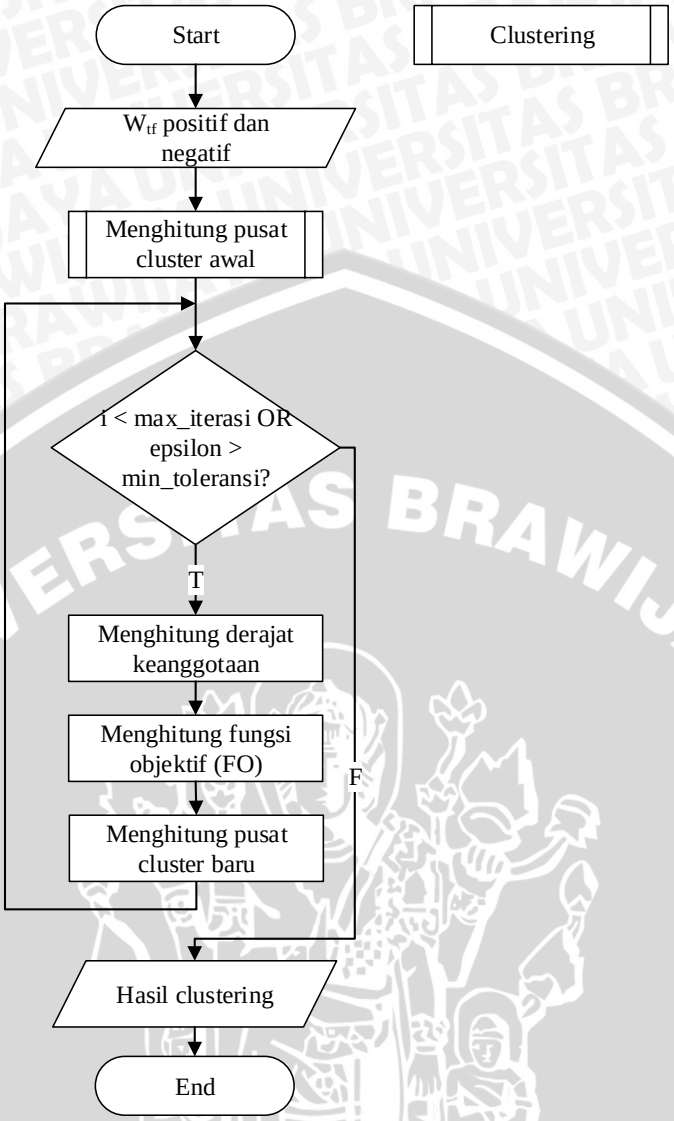
Proses selanjutnya adalah proses *filtering*. *Filtering* bertujuan untuk menghilangkan kata-kata pelengkap yang tidak memiliki nilai terhadap informasi yang disampaikan dokumen. Penghilangan kata-kata tersebut ditentukan berdasarkan daftar *stop word*. Diagram alir filtering ditunjukkan seperti pada Gambar 3.6.



Gambar 3.7 Diagram Alir Stemming Arifin-Setiono
Sumber: [ARI-02]



Gambar 3.8 Diagram Alir Pembobotan Term Frequency



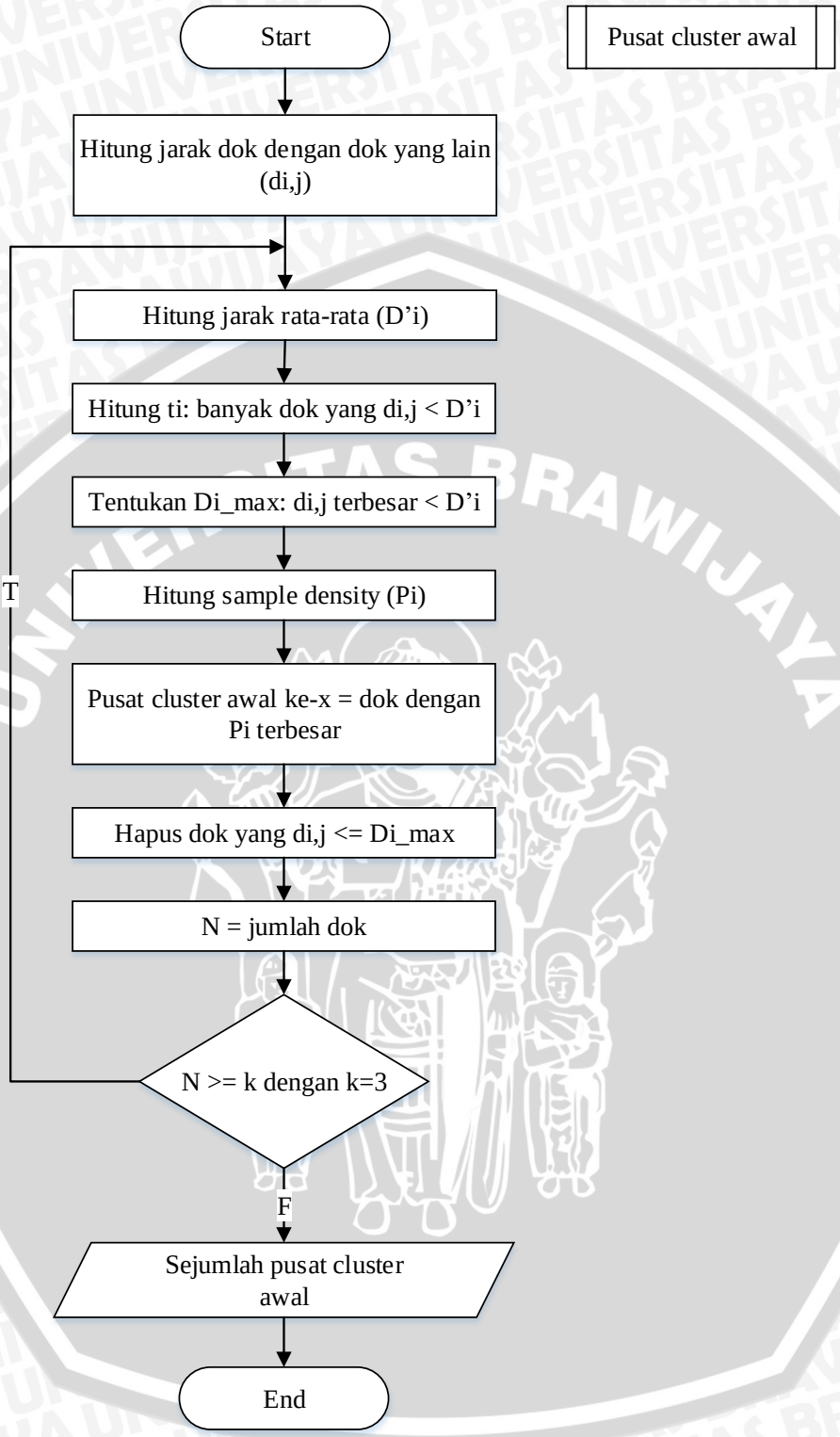
Gambar 3.9 Diagram Alir Improved Fuzzy C-Means
Sumber: [YAN-13]

Dan proses yang terakhir dari *preprocessing* yaitu proses *stemming*, bertujuan untuk mengubah semua kata ke bentuk kata dasarnya. Proses *stemming* dilakukan dengan menggunakan metode *stemming* Arifin-Setiono. Dalam proses *stemming* Arifin-Setiono, semua kata diasumsikan memiliki dua awalan dan tiga akhiran. Setiap kata akan dilakukan pengecekan sebanyak dua kali untuk menghilangkan awalan dan pengecekan sebanyak tiga kali untuk menghilangkan akhiran. Diagram alir *stemming* Arifin-Setiono ditunjukkan seperti pada Gambar 3.7.

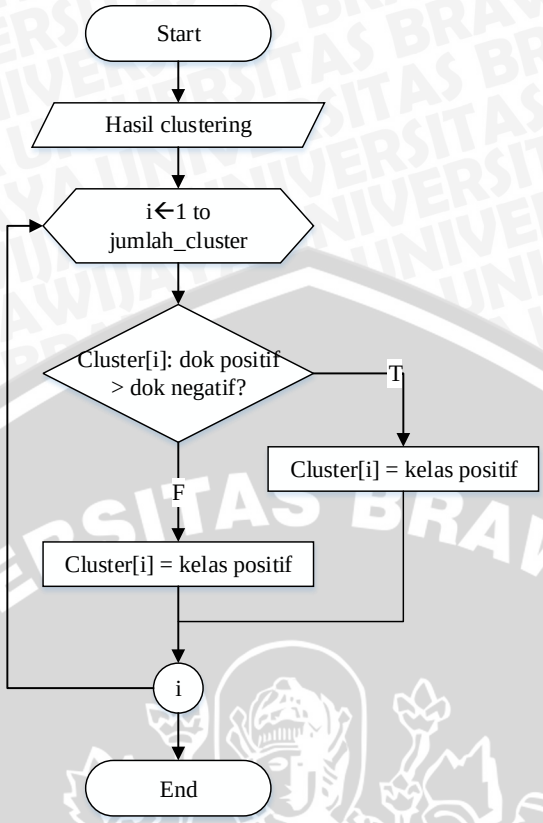
Setelah *preprocessing* selesai, proses selanjutnya adalah pembobotan. Bembobotan dilakukan menggunakan metode pembobotan *term frequency* (W_{tf}). Setiap dokumen dilakukan pembobotan terhadap term positif dan term negatif. Bobot dari term positif dan term negatif ini akan menjadi fitur dari setiap dokumen. Pembobotan dirancang sedemikian hingga mampu menangani kata bernegasi seperti "tidak mengecewakan" atau "tidak jalan". Kata positif yang berpasangan dengan kata negasi seperti kata "tidak" akan ditambahkan bobotnya pada bobot negatif. Begitu juga sebaliknya, kata negatif yang berpasangan dengan kata negasi akan ditambahkan bobotnya pada bobot positif. Proses pembobotan ditunjukkan dalam diagram alir pada Gambar 3.8.

Setelah pembobotan selesai, dokumen siap untuk dilakukan clustering menggunakan Improved Fuzzy C-Means (FCM) dengan proses seperti yang ditunjukkan dalam diagram alir pada Gambar 3.9. Perbaikan (*Improved*) FCM terletak pada proses pemilihan pusat *cluster* awal, dengan menggunakan algoritma tersendiri. Proses pemilihan pusat cluster awal ditunjukkan dalam diagram alir pada Gambar 3.10.

Pada proses tersebut akan menghasilkan beberapa cluster beserta dokumen yang masuk ke dalam cluster tersebut berdasarkan derajat keanggotaan terbesar. Cluster tersebut akan ditentukan kelasnya yaitu positif atau negatif berdasarkan rasio jumlah dokumen di dalamnya. Jika dalam suatu cluster lebih banyak dokumen positif daripada dokumen negatif, maka cluster tersebut merupakan cluster positif. Sebaliknya jika dalam suatu cluster lebih banyak dokumen negatif daripada dokumen positif, maka cluster tersebut merupakan cluster negatif. Proses penentuan kelas cluster ditunjukkan dalam diagram alir pada Gambar 3.11.



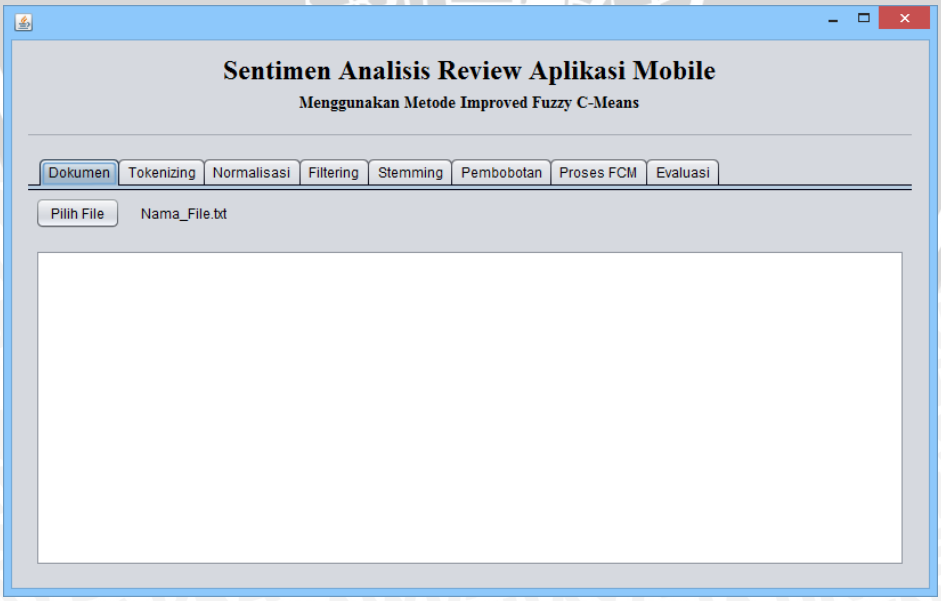
Gambar 3.10 Diagram Alir Pemilihan Pusat Cluster Awal
Sumber: [YAN-13]



Gambar 3.11 Diagram Alir Penentuan Kelas Cluster

3.3.5 Desain Antarmuka

Desain antarmuka bertujuan untuk mempermudah interaksi antar pengguna dan aplikasi. Terdapat dua komponen utama dalam desain antarmuka, yaitu *header* dan *tab menus*.



Gambar 3.12 Perancangan Antarmuka

Header menampilkan informasi utama dari halaman yang aktif. Sedangkan *tab menus* menampilkan menu-menu menuju halaman yang dituju. Terdapat delapan menu yaitu dokumen, *tokenizing*, normalisasi, *filtering*, *stemming*, pembobotan, proses FCM, dan evaluasi. Menu dokumen mengijinkan *user* untuk memilih dokumen *review* yang akan dilakukan analisis sentimen dan menampilkan isi dari dokumen tersebut. Menu *tokenizing*, normalisasi, *filtering*, *stemming*, dan pembobotan menampikan hasil dari proses tersebut terhadap dokumen review. Menu proses FCM menampilkan semua *review* positif dan negatif yang sudah dikelompokkan. Menu evaluasi menampilkan rangkuman hasil *clustering* beserta tingkat keberhasilan aplikasi dalam melakukan *clustering*. Perancangan antarmuka tampilan utama ditunjukkan pada Gambar 3.10.

3.3.6 Manualisasi Analisis Sentimen

Manualisasi berfungsi untuk memberikan gambaran umum perancangan sistem yang akan dibangun. Manualisasi setiap proses analisis sentimen menggunakan Improved Fuzzy C-Means adalah sebagai berikut:

3.3.6.1. Data input dan *preprocessing*

Data input yang akan digunakan dalam sistem berupa *file text document* (.txt). Terdapat 15 data input yang terdiri dari delapan dokumen berlabel negatif dan tujuh dokumen berlabel positif. Data input ditunjukkan pada Tabel 3.1.

Tabel 3.1 Data Input

Nama Dokumen	**Isi Dokumen**
D1	Force close Sangat tidak nyaman Sering force close di android 4.4.2.. Tolong di update
D2	??? Sudah beberapa bulan tidak ada masalah, tapi beberapa hari ini selalu tidak bisa login. Aplikasi nya tdk user friendly seperti mobile banking bank lain, terlalu ribet
D3	Tidak bisa dipakai di 4.4.4 Mohon segera di update untuk Android 4.4.4
D4	Error melulu Dikit2 error minta install ulang, payah banget
D5	Normal Saya sudah pakai normal2 saja, mungkin yang merasa error trus, kalian harus daftar nomor hp dlu di kntor BCA trdekat.. Dan hrus pke internet pket data dr nmor yg sdh d daftar td..god app bca
D6	Good Sangat membantu n sangat berbeda dari aplikasi mobile banking yang lainnya
D7	Berguna Sangat berguna tuk kebutuhan perbankan sehari-hari..

D8	Memudahkan
D9	Good App Sangat membantu...Gak ada error..smpe skr msh berjalan bagus app nya...
D10	sangat menolong & praktis
D11	It's function! Sangat membantu proses mobile sehari2... tanpa harus ribe k atm.. semua tramsaksi dalam genggaman
D12	Gagal update Permintaan update saat buka aplikasi bisa download,tp saat akan di instal tidak bisa.kenapa ga update lewat play store saja? Fix it pls
D13	Beneran nih error 107 trus, payah banget. Bank besar tapi ngurusin masalah aplikasi gini aja ga ditanggapi. Mengecewakan.
D14	Maksa update.... makin jelek tau Nyesel update.. jadi gak bisa sekarang
D15	Apa ini!!?? Diupdate tapi ga bisa diinstall. Jadi ga bisa dipake aplikasinya. Pemaksaan untuk update. Ga ada pilihan close. Cuma ada "ok". Setiap klik OK, download selesai, ngga bisa diinstal, aplikasi langsung tutup. Payah!!!!

Langkah awal yang dilakukan adalah melakukan *preprocessing* terhadap data input tersebut. Proses *preprocessing* terdiri dari *tokenizing, filtering,* dan *stemming*. *Tokenizing* dilakukan dengan cara menghilangkan karakter-karakter pemisah yang bisa berupa spasi, tanda baca, angka, dan karakter selain huruf. Dalam proses ini juga dilakukan pengubahan semua huruf menjadi huruf kecil. Hasil dari proses *tokenizing* ditunjukkan pada Tabel 3.2.

Tabel 3.2 Tokenizing Data Input

Nama Dokumen	**Isi Dokumen**
D1	force close sangat tidak nyaman sering force close di android tolong di update
D2	sudah beberapa bulan tidak ada masalah tapi beberapa hari ini selalu tidak bisa login aplikasi nya tdk user friendly seperti mobile banking bank lain terlalu ribet
D3	tidak bisa dipakai di mohon segera di update untuk android
D4	error melulu dikit error minta install ulang payah banget
D5	normal saya sudah pakai normal saja mungkin yang merasa error trus kalian harus daftar nomor hp dlu di kntor bca trdekat dan hrus pke internet pket data dr nmor yg sdh d daftar td god app bca
D6	good sangat membantu n sangat berbeda dari aplikasi mobile banking yang lainnya
D7	berguna sangat berguna tuk kebutuhan perbankan sehari hari
D8	memudahkan
D9	good app sangat membantu gak ada error smpe skr msh berjalan bagus app nya
D10	sangat menolong praktis

D11	it s function sangat membantu proses mobile sehari tanpa harus ribe k atm semua tramsaksi dalam genggaman
D12	gagal update permintaan update saat buka aplikasi bisa download tp saat akan di instal tidak bisa kenapa ga update lewat play store saja fix it pls
D13	beneran nih error trus payah banget bank besar tapi ngurusin masalah aplikasi gini aja ga ditanggapi mengecewakan
D14	maksa update makin jelek tau nyesel update jadi gak bisa sekarang
D15	apa ini diupdate tapi ga bisa diinstall jadi ga bisa dipake aplikasinya pemaksaan untuk update ga ada pilihan close cuma ada ok setiap klik ok download selesai ngga bisa diinstal aplikasi langsung tutup payah

Tabel 3.3 Data Input Ternormalisasi

Nama Dokumen	**Isi Dokumen**
D1	force close sangat tidak nyaman sering force close di android tolong di update
D2	sudah beberapa bulan tidak ada masalah tapi beberapa hari ini selalu tidak bisa login aplikasi nya tidak user friendly seperti mobile banking bank lain terlalu repot
D3	tidak bisa dipakai di mohon segera di update untuk android
D4	error melulu sedikit error minta install ulang buruk sekali
D5	normal saya sudah pakai normal saja mungkin yang merasa error terus kalian harus daftar nomor hp dahulu di kntor bca terdekat dan harus pakai internet paket data dari nomor yang sudah di daftar tadi god app bca
D6	good sangat membantu dan sangat berbeda dari aplikasi mobile banking yang lainnya
D7	berguna sangat berguna untuk kebutuhan perbankan sehari hari
D8	memudahkan
D9	good app sangat membantu tidak ada error sampai sekarang masih berjalan bagus app nya
D10	sangat menolong praktis
D11	it s function sangat membantu proses mobile sehari tanpa harus repot ke atm semua tramsaksi dalam genggaman
D12	gagal update permintaan update saat buka aplikasi bisa download tapi saat akan di instal tidak bisa kenapa tidak update lewat play store saja fix it pls
D13	benar nih error terus buruk sekali bank besar tapi mengurusi masalah aplikasi begini saja tidak ditanggapi mengecewakan
D14	paksa update semakin jelek tahu menyesal update jadi tidak bisa sekarang
D15	apa ini diupdate tapi tidak bisa diinstall jadi tidak bisa dipakai aplikasinya pemaksaan untuk update tidak ada pilihan close hanya ada ok setiap klik ok download selesai tidak bisa diinstal aplikasi langsung tutup buruk

Data input terlebih dahulu akan dinormalisasi dengan mengubah kata yang tidak baku menjadi kata baku. Hal ini dilakukan agar dalam proses selanjutnya, semua kata bisa diproses dengan benar ketika dibandingkan dengan kata dalam kamus, baik kamus *stoplist* maupun kamus kata dasar. Data input yang ternormalisasi ditunjukkan pada Tabel 3.3.

Setelah proses normalisasi selesai, selanjutnya adalah *filtering*. *Filtering* bertujuan untuk menghilangkan kata-kata tidak penting dengan membandingkan dengan kata-kata dalam kamus *stoplist*. Hasil dari *filtering* ditunjukkan pada Tabel 3.4.

Tabel 3.4 Filtering Data Input

Nama Dokumen	**Isi Dokumen**
D1	force close tidak nyaman force close android tolong update
D2	tidak masalah tidak bisa login aplikasi tidak user friendly mobile banking bank repot
D3	tidak bisa dipakai mohon update android
D4	error error minta install ulang buruk
D5	normal normal error daftar nomor hp kntor bca terdekat internet paket data nomor daftar god app bca
D6	good membantu berbeda aplikasi mobile banking lainnya
D7	berguna berguna kebutuhan perbankan
D8	memudahkan
D9	good app membantu tidak error berjalan bagus app
D10	menolong praktis
D11	it s function membantu proses mobile tanpa repot atm tramsaksi genggaman
D12	gagal update permintaan update buka aplikasi bisa download instal tidak bisa tidak update lewat play store fix it pls
D13	benar error buruk bank besar ngurus masalah aplikasi tidak ditanggapi mengecewakan
D14	paksa update jelek menyesal update tidak bisa
D15	diupdate tidak bisa diinstall tidak bisa dipakai aplikasinya pemaksaan update tidak pilihan close ok klik ok download selesai tidak bisa diinstal aplikasi tutup buruk

Proses yang terakhir yaitu *stemming*, bertujuan untuk mengubah semua kata ke bentuk kata dasarnya dengan menggunakan metode *stemming* Arifin-Setiono. Hasil dari proses *stemming* ditunjukkan pada Tabel 3.5.

Tabel 3.5 Stemming Data Input

Nama Dokumen	Isi Dokumen
D1	force close tidak nyaman force close android tolong update
D2	tidak masalah tidak bisa log aplikasi tidak user friendly mobile banking bank lain repot
D3	tidak bisa pakai mohon update android
D4	error error minta install ulang buruk
D5	normal normal error daftar nomor hp kntor bca dekat internet paket data nomor daftar god app bca
D6	good bantu beda aplikasi mobile banking lain
D7	guna guna butuh bank
D8	mudah
D9	good app bantu tidak error jalan bagus app
D10	tolong praktis
D11	It s function bantu proses mobile tanpa repot atm transaksi genggam
D12	gagal update minta update buka aplikasi bisa download instal tidak bisa tidak update lewat play store fix it pls
D13	benar error buruk bank besar masalah aplikasi tidak tanggap kecewa
D14	paksa update jelek kesal update tidak bisa
D15	update tidak bisa install tidak bisa pakai aplikasi paksa update tidak pilih close cuma ok klik ok download selesai tidak bisa instal aplikasi tutup buruk

3.3.6.2. Pembobotan dokumen

Setelah *preprocessing* selesai, langkah selanjutnya adalah melakukan pembobotan menggunakan metode pembobotan *term frequency* (W_{tf}) terhadap term positif dan negatif. Dokumen yang memiliki nilai W_{tf} positif lebih besar dari W_{tf} negatif akan diberikan label positif. Sebaliknya, dokumen yang memiliki nilai W_{tf} negatif lebih besar dari W_{tf} positif akan diberikan label negatif. Pembobotan dokumen dihitung berdasarkan Persamaan (2-1) dan hasilnya ditunjukkan pada Tabel 3.6 hingga Tabel 3.20. Contoh pembobotan dokumen D1 untuk sentimen negatif sebagai berikut:

$$w_{tf-neg} = 1 + \log TF = 1 + \log 1 = 1 + 0 = 1$$

Tabel 3.6 Pembobotan Dokumen D1

Dokumen D1		
Term	**Positif**	**Negatif**
force		
close		
tidak nyaman		1

force		
close		
android		
tolong		
update		
TF	**0**	**1**
Wtf	**0**	**1**
Label	**negatif**	

Tabel 3.7 Pembobotan Dokumen D2

Dokumen D2		
Term	**Positif**	**Negatif**
tidak masalah	1	
tidak bisa		1
log		
aplikasi		
tidak		
user		
friendly		
mobile		
banking		
bank		
lain		
repot		1
TF	**1**	**2**
Wtf	**1**	**1,301**
Label	**negatif**	

Tabel 3.8 Pembobotan Dokumen D3

Dokumen D3		
Term	**Positif**	**Negatif**
tidak bisa		1
pakai		
mohon		1
update		
android		
TF	**0**	**2**
Wtf	**0**	**1,301**
Label	**negatif**	

Tabel 3.9 Pembobotan Dokumen D4

Dokumen D4		
Term	**Positif**	**Negatif**
error		
error		
minta		

install		
ulang		
buruk		1
TF	**0**	**1**
Wtf	**0**	**1**
Label	**negatif**	

Tabel 3.10 Pembobotan Dokumen D5

Dokumen D5		
Term	**Positif**	**Negatif**
normal	1	
normal	1	
error		
daftar		
nomor		
hp		
kntor		
bca		
dekat		
internet		
paket		
data		
nomor		
daftar		
god		
app		
bca		
TF	**2**	**0**
Wtf	**1,301**	**0**
Label	**positif**	

Tabel 3.11 Pembobotan Dokumen D6

Dokumen D6		
Term	**Positif**	**Negatif**
good		
bantu	1	
beda		
aplikasi		
mobile		
banking		
lain		
TF	**1**	**0**
Wtf	**1**	**0**
Label	**positif**	

Tabel 3.12 Pembobotan Dokumen D7

Dokumen D7		
Term	**Positif**	**Negatif**
guna	1	
guna	1	
butuh		
bank		
TF	**2**	**0**
Wtf	**1,301**	**0**
Label	**negatif**	

Tabel 3.13 Pembobotan Dokumen D8

Dokumen D8		
Term	**Positif**	**Negatif**
mudah	1	
TF	**1**	**0**
Wtf	**1**	**0**
Label	**positif**	

Tabel 3.14 Pembobotan Dokumen D9

Dokumen D9		
Term	**Positif**	**Negatif**
good		
app		
bantu	1	
tidak		
error		
jalan		
bagus	1	
app		
TF	**2**	**0**
Wtf	**1,301**	**0**
Label	**positif**	

Tabel 3.15 Pembobotan Dokumen D10

Dokumen D10		
Term	**Positif**	**Negatif**
Tolong		
Praktis	1	
TF	**1**	**0**
Wtf	**1**	**0**
Label	**positif**	

Tabel 3.16 Pembobotan Dokumen D11

Dokumen D11		
Term	**Positif**	**Negatif**

it		
s		
function		
bantu	1	
proses		
mobile		
tanpa repot	1	
atm		
transaksi		
genggam		
TF	**2**	**0**
Wtf	**1,301**	**0**
Label	**positif**	

Tabel 3.17 Pembobotan Dokumen D12

Dokumen D12		
Term	**Positif**	**Negatif**
gagal		1
update		
minta		
update		
buka		
aplikasi		
bisa	1	
download		
instal		
tidak bisa		1
tidak		
update		
lewat		
play		
store		
fix		
it		
pls		
TF	**1**	**2**
Wtf	**1**	**1,301**
Label	**negatif**	

Tabel 3.18 Pembobotan Dokumen D13

Dokumen D13		
Term	**Positif**	**Negatif**
benar		
error		
buruk		1
bank		

besar		
masalah		1
aplikasi		
tidak tanggap		1
kecewa		1
TF	**0**	**4**
Wtf	**0**	**1,602**
Label	**negatif**	

Tabel 3.19 Pembobotan Dokumen D14

Dokumen D14		
Term	**Positif**	**Negatif**
paksa		
update		
jelek		1
kesal		1
update		
tidak bisa		1
TF	**0**	**3**
Wtf	**0**	**1,477**
Label	**negatif**	

Tabel 3.20 Pembobotan Dokumen D15

Dokumen D15		
Term	**Positif**	**Negatif**
update		
tidak bisa		1
install		
tidak bisa		1
pakai		
aplikasi		
paksa		
update		
tidak		
pilih		
close		
cuma		
ok		
klik		
ok		
download		
selesai		
tidak bisa		1
instal		
aplikasi		
tutup		

buruk		1
TF	**0**	**4**
Wtf	**0**	**1,602**
Label	**negatif**	

3.3.6.3. Clustering dokumen

Setelah dokumen dilakukan pembobotan, bobot positif dan negatif tersebut akan menjadi fitur dalam melakukan clustring. *Clustering* dokumen menggunakan metode Improved Fuzzy C-Means *(FCM)*. Bobot *term* positif dan *term* negatif pada proses sebelumnya akan menjadi fitur setiap dokumen dalam *clustering*. Bobot dari semua dokumen dirangkum pada Tabel 3.21. Tahapan dalam Improved Fuzzy C-Means adalah sebagai berikut:

1. Pemilihan pusat *cluster* awal

Sebelum masuk ke proses utama, ada beberapa parameter yang harus ditentukan di awal proses perhitungan. Parameter tersebut antara lain: pembobot (m) = 2, maksimum iterasi = 100, dan toleransi penghentian = 1.10^{-4}.

Setelah itu masuk ke proses perhitungan jarak antar dokumen. Perhitungan jarak dilakukan menggunakan metode *Euclidean distance* menggunakan Persamaan (2-4). Jarak antar semua dokumen ditunjukkan pada Tabel 3.22. Contoh perhitungan jarak antara D1 dengan D2 adalah sebagai berikut:

$$d_{1,2} = \sqrt{(w_{tf-pos}D1 - w_{tf-pos}D2)^2 + (w_{tf-neg}D1 - w_{tf-neg}D2)^2}$$

$$d_{1,2} = \sqrt{(0-1)^2 + (1-1{,}301)^2} = 1{,}044$$

Setelah mendapatkan semua jarak antar dokumen, langkah selanjutnya adalah mencari nilai $D'i$. $D'i$ adalah nilai rata-rata dari total jarak semua dokumen, yang dihitung menggunakan Persamaan (2-7). Contoh perhitungan $D'1$ ($D'i$ dari D1) sebagai berikut:

$$D'1 = \frac{d_{1,1} + d_{1,2} + d_{1,3} + \cdots + d_{1,15}}{n-1}$$

$$D'1 = \frac{0 + 1{,}044 + \cdots + 0{,}477 + 0{,}477}{15-1} = \frac{14{,}88}{14} = 1{,}063$$

Langkah selanjutnya adalah mencari nilai Pi menggunakan Persamaan (2-8). Dalam persamaan tersebut, ti adalah banyaknya obyek yang jaraknya kurang dari $D'i$. Dengan nilai $D'1$ = 1,063, maka nilai t1 = 8. Delapan dokumen tersebut

adalah D1, D2, D3, D4, D12, D13, D14, dan D15. Sedangkan D1-max adalah jarak terbesar dari sekian dokumen yang jaraknya kurang dari $D'1$. Dari delapan dokumen tersebut, dokumen yang memiliki jarak terbesar dari D1 adalah D2 dan D12 dengan nilai 1,044. Sehingga nilai dari D1-max = 1,044. Dan perhitungan P1 (Pi dari D1) adalah sebagai berikut:

$$P1 = \frac{t1}{D1 - max} = \frac{8}{1{,}044} = 7{,}66$$

Semua dokumen dihitung nilai Pi-nya. Setelah mendapatkan nilai Pi semua dokumen, dokumen yang menjadi pusat cluster awal adalah dokumen yang memiliki nilai Pi terbesar. Ada tujuh dokumen yang memiliki nilai Pi terbesar yaitu D5, D6, D7, D8, D9, D10, dan D11. Dari tujuh dokumen tersebut, dipilih sembarang dokumen, semisal D5 sebagai pusat cluster awal ke-1 (C1). Nilai dari $D'i$, ti, Di-max, dan Pi dari semua dokumen dalam perhitungan mendapatkan C1 ditunjukkan pada Tabel 3.23.

Untuk mendapatkan pusat cluster awal berikutnya, beberapa dokumen akan tidak diikutsertakan dalam perhitungan. Dokumen yang tidak diikutsertakan adalah dokumen yang memiliki jarak terhadap C1 kurang dari atau sama dengan Di-max D5 (D5-max). Dokumen tersebut antara lain: D5, D6, D7, D8, D9, D10, dan D11. Sehingga, dokumen yang akan diproses dalam perhitungan mendapatkan C2 antara lain: D1, D2, D3, D4, D12, D13, D14, dan D15. Dokumen tersebut dihitung jaraknya dan hasilnya seperti pada Tabel 3.24, untuk proses perhitungan pusat *cluster* awal selanjutnya.

Langkah tersebut di atas diulangi sedemikian hingga jumlah dokumen yang tersisa kurang dari batas minimal yaitu tiga dokumen. Hingga kondisi tersebut memenuhi, telah ditemukan sebanyak dua pusat cluster awal. Tahapan mendapatkan pusat cluster awal ke-2 (C2) ditunjukkan pada Tabel 3.25. Dan jarak dokumen terhadap pusat *cluster* awal C1 dan C2 ditunjukkan pada Tabel 3.26.

2. Menghitung derajat keanggotaan awal

Setelah mendapatkan pusat *cluster* awal, proses selanjutnya adalah menghitung derajat keanggotaan awal semua dokumen terhadap semua pusat *cluster* menggunakan Persamaan (2-3). Derajat keanggotaan semua dokumen terhadap semua cluster ditunjukkan pada Tabel 3.27 dan Tabel 3.28. Contoh

perhitungan derajat keanggotaan D4 terhadap cluster C1 ($u_{4,1}$) adalah sebagai berikut:

$$u_{4,1} = \frac{1}{\left(\frac{d_{4,1}}{d_{1,1}}\right)^2 + \left(\frac{d_{4,1}}{d_{2,1}}\right)^2} = \frac{1}{\left(\frac{1{,}641}{0}\right)^2 + \left(\frac{1{,}641}{1{,}84}\right)^2}$$

$$= \frac{1}{0 + 0{,}795} = 1{,}257$$

Dokumen yang menjadi pusat cluster ke-x, nilai derajat keanggotaannya terhadap terhadap *cluster* ke-x secara otomatis adalah satu, sedangkan nilai derajat keanggotaannya terhadap *cluster* selain *cluster* ke-x secara otomatis bernilai nol. Seperti pada Tabel 3.27, dokumen D5 menjadi pusat *cluster* awal C1, dan beberapa dokumen yang lain memiliki nilai fitur yang sama dengan D5 yaitu D7, D9 dan D11. Maka ketiga dokumen tersebut juga memiliki derajat keanggotaan bernilai satu terhadap C1, dan bernilai nol terhadap C2.

Setelah didapatkan nilai derajat keanggotaan, nilai tersebut dilakukan normalisasi antara derajat keanggotaan yang satu dengan yang lain. Sebab total semua derajat keanggotaan untuk tiap dokumen bernilai satu. Derajat keanggotaan yang ternormalisasi dan nilai kuadratnya ditunjukkan pada Tabel 3.29. Contoh perhitungan normalisasi derajat keanggotaan D1 terhadap C1 sebagai berikut:

$$norm_u1{,}1 = \frac{1{,}257}{1{,}257 + 37{,}36} = 0{,}033$$

Proses selanjutnya yaitu menghitung fungsi obyektif, yang berfungsi untuk mengetahui nilai *error* dari hasil clustering sementara menggunakan Persamaan (2-6). Proses mendapatkan nilai fungsi obyektif ditunjukkan pada Tabel 3.30. Contoh perhitungan fungsi obyektif sebagai berikut:

$$F.O.1 = {d_{1,1}}^2 * {u_{1,1}}^2 + \cdots + {d_{15,1}}^2 * {u_{15,1}}^2 + {d_{1,2}}^2 * {u_{1,2}}^2 + \cdots + {d_{15,2}}^2 * {u_{15,2}}^2$$

$$= 1{,}641^2 * 0{,}033^2 + \cdots + 2{,}064^2 * 0{,}021^2 + 0{,}301^2 * 0{,}967^2$$

$$+ \cdots + 0{,}301^2 * 0{,}979^2 = 0{,}003 + \cdots + 0{,}002 + 0{,}085 + 0{,}087$$

$$= 1{,}928$$

Tabel 3.21 Fitur Dokumen

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
wtf-pos ($x_{i,1}$)	0	1	0	0	1,301	1	1,301	1	1,301	1	1,301	1	0	0	0
wtf-neg ($x_{i,2}$)	1	1,301	1,301	1	0	0	0	0	0	0	0	1,301	1,602	1,477	1,602

Tabel 3.22 Jarak Antar Dokumen

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
D1	0	1,044	0,301	0	1,641	1,414	1,641	1,414	1,641	1,414	1,641	1,044	0,602	0,477	0,602
D2	1,044	0	1	1,044	1,335	1,301	1,335	1,301	1,335	1,301	1,335	0	1,044	1,015	1,044
D3	0,301	1	0	0,301	1,84	1,641	1,84	1,641	1,84	1,641	1,84	1	0,301	0,176	0,301
D4	0	1,044	0,301	0	1,641	1,414	1,641	1,414	1,641	1,414	1,641	1,044	0,602	0,477	0,602
D5	1,641	1,335	1,84	1,641	0	0,301	0	0,301	0	0,301	0	1,335	2,064	1,968	2,064
D6	1,414	1,301	1,641	1,414	0,301	0	0,301	0	0,301	0	0,301	1,301	1,889	1,784	1,889
D7	1,641	1,335	1,84	1,641	0	0,301	0	0,301	0	0,301	0	1,335	2,064	1,968	2,064
D8	1,414	1,301	1,641	1,414	0,301	0	0,301	0	0,301	0	0,301	1,301	1,889	1,784	1,889
D9	1,641	1,335	1,84	1,641	0	0,301	0	0,301	0	0,301	0	1,335	2,064	1,968	2,064
D10	1,414	1,301	1,641	1,414	0,301	0	0,301	0	0,301	0	0,301	1,301	1,889	1,784	1,889
D11	1,641	1,335	1,84	1,641	0	0,301	0	0,301	0	0,301	0	1,335	2,064	1,968	2,064
D12	1,044	0	1	1,044	1,335	1,301	1,335	1,301	1,335	1,301	1,335	0	1,044	1,015	1,044
D13	0,602	1,044	0,301	0,602	2,064	1,889	2,064	1,889	2,064	1,889	2,064	1,044	0	0,125	0
D14	0,477	1,015	0,176	0,477	1,968	1,784	1,968	1,784	1,968	1,784	1,968	1,015	0,125	0	0,125
D15	0,602	1,044	0,301	0,602	2,064	1,889	2,064	1,889	2,064	1,889	2,064	1,044	0	0,125	0

Tabel 3.23 Pemilihan Pusat Cluster Awal Ke-1

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
D'i	1,063	1,103	1,119	1,063	1,057	0,988	1,057	0,988	1,057	0,988	1,057	1,103	1,26	1,188	1,26
Ti	8	8	8	8	7	7	7	7	7	7	7	8	8	8	8
Di-max	1,044	1,044	1	1,044	0,301	0,301	0,301	0,301	0,301	0,301	0,301	1,044	1,044	1,015	1,044
Pi	7,66	7,66	8	7,66	23,25	23,25	23,25	23,25	23,25	23,25	23,25	7,66	7,66	7,879	7,66
C1	**23,25 (D5)**														

Tabel 3.24 Jarak Antar Dokumen Yang Tersisa Setelah Mendapatkan C1

	D1	D2	D3	D4	D12	D13	D14	D15
D1	0	1,044	0,301	0	1,044	0,602	0,477	0,602
D2	1,044	0	1	1,044	0	1,044	1,015	1,044
D3	0,301	1	0	0,301	1	0,301	0,176	0,301
D4	0	1,044	0,301	0	1,044	0,602	0,477	0,602
D12	1,044	0	1	1,044	0	1,044	1,015	1,044
D13	0,602	1,044	0,301	0,602	1,044	0	0,125	0
D14	0,477	1,015	0,176	0,477	1,015	0,125	0	0,125
D15	0,602	1,044	0,301	0,602	1,044	0	0,125	0

Tabel 3.25 Pemilihan Pusat Cluster Awal Ke-2

	D1	D2	D3	D4	D12	D13	D14	D15
D'i	0,582	0,885	0,483	0,582	0,885	0,531	0,487	0,531
Ti	4	2	6	4	2	4	6	4
Di-max	0,477	0	0,301	0,477	0	0,301	0,477	0,301
Pi	8,384	0	19,93	8,384	0	13,29	12,58	13,29
C2	**19,93 (D3)**							

Tabel 3.26 Jarak Dokumen Terhadap Pusat Cluster Awal

	C1	C2
D1	1,641	0,301
D2	1,335	1
D3	1,84	0
D4	1,641	0,301
D5	0	1,84
D6	0,301	1,641
D7	0	1,84
D8	0,301	1,641
D9	0	1,84
D10	0,301	1,641
D11	0	1,84
D12	1,335	1
D13	2,064	0,301
D14	1,968	0,176
D15	2,064	0,301
C1	0	1,84
C2	1,84	0

Tabel 3.27 Derajat Keanggotaan Terhadap C1

Di	Dk		ui1
	1	2	
1	0	0,795	1,257
2	0	0,527	1,898
3	-	-	0
4	0	0,795	1,257
5	-	-	1
6	0	0,027	37,36
7	-	-	1
8	0	0,027	37,36
9	-	-	1
10	0	0,027	37,36
11	-	-	1
12	0	0,527	1,898

13	0	1,258	0,795
14	0	1,145	0,874
15	0	1,258	0,795

Tabel 3.28 Derajat Keanggotaan Terhadap C2

Di	**Dk**		**ui2**
	1	**2**	
1	0,027	0	37,36
2	0,295	0	3,385
3	-	-	1
4	0,027	0	37,36
5	-	-	0
6	0,795	0	1,257
7	-	-	0
8	0,795	0	1,257
9	-	-	0
10	0,795	0	1,257
11	-	-	0
12	0,295	0	3,385
13	0,027	0	37,36
14	0,009	0	109,2
15	0,027	0	37,36

Tabel 3.29 Normalisasi Dan Kuadrat Derajat Keanggotaan

	norm_ui1	**$ui1^2$**	**norm_ui2**	**$ui2^2$**
D1	0,033	0,001	0,967	0,936
D2	0,359	0,129	0,641	0,411
D3	0	0	1	1
D4	0,033	0,001	0,967	0,936
D5	1	1	0	0
D6	0,967	0,936	0,033	0,001
D7	1	1	0	0
D8	0,967	0,936	0,033	0,001
D9	1	1	0	0
D10	0,967	0,936	0,033	0,001
D11	1	1	0	0
D12	0,359	0,129	0,641	0,411
D13	0,021	4E-04	0,979	0,959
D14	0,008	6E-05	0,992	0,984
D15	0,021	4E-04	0,979	0,959
∑		**7,069**		**6,598**

Tabel 3.30 Fungsi Obyektif

Proses FO		
1	0,003	0,085
2	0,23	0,411
3	0	0
4	0,003	0,085
5	0	0
6	0,085	0,003
7	0	0
8	0,085	0,003
9	0	0
10	0,085	0,003
11	0	0
12	0,23	0,411
13	0,002	0,087
14	2E-04	0,031
15	0,002	0,087
∑	0,724	1,204
∑	**F.O.1**	**1,928**

Sampai pada tahap ini dapat diketahui hasil sementara clustering seperti pada Tabel 3.31. Dokumen dengan nilai derajat keanggotaan terbesar pada *cluster* tertentu dipilih sebagai hasil *clustering.*

Tabel 3.31 Hasil Clustering

	Hasil
D1	2
D2	2
D3	2
D4	2
D5	1
D6	1
D7	1
D8	1
D9	1
D10	1
D11	1
D12	2
D13	2
D14	2

D15	2

3. Menghitung pusat cluster baru

Proses selanjutnya adalah menghitung pusat *cluster* baru menggunakan Persamaan (2-5). Proses dan hasil pusat *cluster* baru ditunjukkan pada Tabel 3.32. Contoh perhitungan mendapatkan pusat *cluster* C1 baru untuk fitur bobot positif (wtf-pos) adalah sebagai berikut:

$$wtf - pos\ C1 = \frac{{u_{1,1}}^2 * x_{1,1} + {u_{2,1}}^2 * x_{2,1} + \cdots + {u_{15,1}}^2 * x_{15,1}}{{u_{1,1}}^2 + {u_{2,1}}^2 + \cdots + {u_{15,1}}^2}$$

$$= \frac{0{,}033^2 * 0 + 0{,}359^2 * 1 + \cdots + 0{,}021^2 * 0}{0{,}033^2 + 0{,}359^2 + \cdots + 0{,}021^2} = \frac{8{,}27}{7{,}069} = 1{,}17$$

Tabel 3.32 Pusat Cluster Baru

	C1 baru		C2 baru	
	wtf-pos	**wtf-neg**	**wtf-pos**	**wtf-neg**
1	0	0,001	0	0,936
2	0,129	0,168	0,411	0,534
3	0	0	0	1,301
4	0	0,001	0	0,936
5	1,301	0	0	0
6	0,936	0	0,001	0
7	1,301	0	0	0
8	0,936	0	0,001	0
9	1,301	0	0	0
10	0,936	0	0,001	0
11	1,301	0	0	0
12	0,129	0,168	0,411	0,534
13	0	7E-04	0	1,536
14	0	9E-05	0	1,454
15	0	7E-04	0	1,536
∑	**8,27**	**0,339**	**0,824**	**8,767**
C	**1,17**	**0,048**	**0,125**	**1,329**

4. Menghitung derajat keanggotaan baru

Setelah mendapatkan nilai pusat *cluster* baru, proses selanjutnya adalah menghitung derajat keanggotaan baru. Tapi sebelum itu, harus dihitung terlebih dahulu jarak semua dokumen terhadap semua pusat *cluster* baru menggunakan Persamaan (2-4). Jarak semua dokumen terhadap semua pusat *cluster* baru

ditunjukkan pada Tabel 3.33. Derajat keanggotaan semua dokumen terhadap semua cluster baru beserta derajat keanggotaan ternormalisasi ditunjukkan pada Tabel 3.34 hingga Tabel 3.36.

Tabel 3.33 Jarak Dokumen Terhadap Pusat Cluster Baru

	C1	C2
D1	1,508	0,352
D2	1,264	0,876
D3	1,714	0,128
D4	1,508	0,352
D5	0,14	1,774
D6	0,177	1,591
D7	0,14	1,774
D8	0,177	1,591
D9	0,14	1,774
D10	0,177	1,591
D11	0,14	1,774
D12	1,264	0,876
D13	1,945	0,301
D14	1,847	0,194
D15	1,945	0,301
C1	0	1,653
C2	1,653	0

Tabel 3.34 Derajat Keanggotan Terhadap C1 Baru

Di	**Dk**		**ui1**
	1	**2**	
1	0	0,833	1,201
2	0	0,585	1,709
3	0	1,076	0,93
4	0	0,833	1,201
5	0	0,007	140,1
6	0	0,011	87,64
7	0	0,007	140,1
8	0	0,011	87,64
9	0	0,007	140,1
10	0	0,011	87,64
11	0	0,007	140,1
12	0	0,585	1,709
13	0	1,385	0,722
14	0	1,248	0,801
15	0	1,385	0,722

Tabel 3.35 Derajat Keanggotan Terhadap C2 Baru

Di	Dk		ui2
	1	2	
1	0,045	0	22,09
2	0,281	0	3,564
3	0,006	0	166,9
4	0,045	0	22,09
5	1,152	0	0,868
6	0,926	0	1,079
7	1,152	0	0,868
8	0,926	0	1,079
9	1,152	0	0,868
10	0,926	0	1,079
11	1,152	0	0,868
12	0,281	0	3,564
13	0,033	0	30,26
14	0,014	0	72,63
15	0,033	0	30,26

Tabel 3.36 Normalisasi Dan Kuadrat Derajat Keanggotaan Baru

	norm_ui1	$ui1^2$	norm_ui2	$ui2^2$
D1	0,052	0,003	0,948	0,9
D2	0,324	0,105	0,676	0,457
D3	0,006	3E-05	0,994	0,989
D4	0,052	0,003	0,948	0,9
D5	0,994	0,988	0,006	4E-05
D6	0,988	0,976	0,012	1E-04
D7	0,994	0,988	0,006	4E-05
D8	0,988	0,976	0,012	1E-04
D9	0,994	0,988	0,006	4E-05
D10	0,988	0,976	0,012	1E-04
D11	0,994	0,988	0,006	4E-05
D12	0,324	0,105	0,676	0,457
D13	0,023	5E-04	0,977	0,954
D14	0,011	1E-04	0,989	0,978
D15	0,023	5E-04	0,977	0,954
∑		**7,095**		**6,589**

Hingga pada proses ini didapatkan nilai epsilon 0,257 seperti yang ditunjukkan pada Tabel 3.37. Nilai epsilon didapatkan dari selisih fungsi obyektif sekarang dengan yang sebelumnya (F.O.2 – F.O.1).

Tabel 3.37 Fungsi Obyektif Baru

Proses FO		
1	0,006	0,111
2	0,168	0,35
3	9E-05	0,016
4	0,006	0,111
5	0,019	1E-04
6	0,03	4E-04
7	0,019	1E-04
8	0,03	4E-04
9	0,019	1E-04
10	0,03	4E-04
11	0,019	1E-04
12	0,168	0,35
13	0,002	0,086
14	4E-04	0,037
15	0,002	0,086
∑	0,521	1,15
∑	**F.O.2**	**1,671**
	Epsilon	**0,257**

Langkah no. 3 dan no. 4 di atas dilakukan berulang-ulang hingga memenuhi maksimum iterasi atau nilai epsilon kurang dari atau sama dengan toleransi penghentian yang sudah ditentukan sebelumnya. Hingga pada iterasi ke-5 didapatkan nilai epsilon 0,000012 seperti pada Tabel 3.38 dan dengan hasil akhir *clustering* seperti ditunjukkan pada Tabel 3.39.

Tabel 3.38 Hasil Clustering Baru

Proses FO		
1	0,007	0,118
2	0,157	0,349
3	2E-04	0,021
4	0,007	0,118
5	0,018	1E-04
6	0,029	4E-04
7	0,018	1E-04
8	0,029	4E-04
9	0,018	1E-04
10	0,029	4E-04
11	0,018	1E-04

12	0,157	0,349
13	0,002	0,088
14	5E-04	0,04
15	0,002	0,088
∑	0,495	1,173
∑	**F.O.5**	**1,668**
	Epsilon	**0,000012**

Tabel 3.39 Hasil Clustering Akhir

	Hasil
D1	2
D2	2
D3	2
D4	2
D5	1
D6	1
D7	1
D8	1
D9	1
D10	1
D11	1
D12	2
D13	2
D14	2
D15	2

Dari hasil akhir *clustering*, ada tujuh dokumen masuk *cluster* 1, delapan dokumen masuk *cluster* 2. Dari sembilan dokumen pada *cluster* 1 tersebut, semua dokumen berlabel positif. Sehingga cluster 1 adalah *cluster* positif. Dari delapan dokumen pada *cluster* 2, semua dokumen berlabel negatif. Sehingga *cluster* 2 adalah *cluster* negatif.

3.3.6.4. Evaluasi

Hasil akhir clustering yang telah didapatkan akan dibandingkan dengan nilai aktual untuk mengetahui keefektifan dari sistem. Hasil akhir clustering dan nilai aktual ditunjukkan pada Tabel 3.41. Untuk mengukur keefektifan, dapat dilihat dari nilai *precision*, *recall*, dan *f-measure*.

Tabel 3.40 Hasil Clustering Akhir

	Hasil Sistem	Nilai Aktual
D1	negatif	negatif
D2	negatif	negatif
D3	negatif	negatif
D4	negatif	negatif
D5	positif	positif
D6	positif	positif
D7	positif	positif
D8	positif	positif
D9	positif	positif
D10	positif	positif
D11	positif	positif
D12	negatif	negatif
D13	negatif	negatif
D14	negatif	negatif
D15	negatif	negatif

Perhitungan nilai *precision*, *recall*, dan *f-measure* berdasarkan Persamaan (2-9) hingga Persamaan (2-11) adalah sebagai berikut:

- Untuk kelas positif:

$$Precision = \frac{TP}{TP + FP} = \frac{7}{7 + 0} = 1 = 100\%$$

$$Recall = \frac{TP}{TP + FN} = \frac{7}{7 + 0} = 1 = 100\%$$

$$F - Measure = \frac{2\ x\ 1\ x\ 1}{1 + 1} = 1 = 100\%$$

- Untuk kelas negatif:

$$Precision = \frac{TP}{TP + FP} = \frac{8}{8 + 0} = 1 = 100\%$$

$$Recall = \frac{TP}{TP + FN} = \frac{8}{8 + 0} = 1 = 100\%$$

$$F - Measure = \frac{2\ x\ 1\ x\ 1}{1 + 1} = 1 = 100\%$$

- Rata-rata:

$$Precision = \frac{100\% + 100\%}{2} = 100\%$$

$$Recall = \frac{100\% + 100\%}{2} = 100\%$$

$$F - Measure = \frac{100\% + 100\%}{2} = 100\%$$

Didapatkan nilai rata-rata *precision* 100%, *recall* 100%, dan *f-measure* 100%. Hal ini menunjukkan bahwa metode Improved Fuzzy C-Means efektif untuk menangani permasalahan analisis sentimen.

3.4 Implementasi

Implementasi aplikasi analisis sentimen *review* aplikasi *mobile* menggunakan metode Improve Fuzzy C-Means dilakukan dengan mengacu pada perancangan sistem. Implementasi perangkat lunak dilakukan menggunakan bahasa pemrograman JAVA. Implementasi aplikasi meliputi:

1. Struktur data yang digunakan meliputi Array, ArrayList, dan HashMap.
2. Pembuatan *user interface* berupa halaman-halaman yang menerima input dari user dan menampilkan proses di dalamnya.
3. Melakukan *preprocessing* data input.
4. Melakukan pembobotan menggunakan metode pembobotan *term frequency.*
5. Perhitungan *clustering* menggunakan metode Improved Fuzzy C-Means.
6. Menghasilkan keluaran berupa kelompok sentimen dari data input.

3.5 Pengujian

Pengujian perangkat lunak pada skripsi ini dilakukan agar dapat menunjukkan bahwa perangkat lunak telah mampu bekerja sesuai dengan kebutuhan yang ditentukan. Pengujian yang dilakukan meliputi:

1. Pengukuran *precision* yang mengacu pada Persamaan (2-9)
2. Pengukuran *recall* yang mengacu pada Persamaan (2-10)
3. Pengukuran *f-measure* yang mengacu pada Persamaan (2-11)

3.5.1 Skenario Pengujian

Skenario pengujian yang akan dilakukan antara lain: pengujian terhadap parameter pembobot yang beragam dan pengujian terhadap variasi jumlah dokumen positif dan dokumen negatif.

1. Pengujian terhadap variasi parameter pembobot dari Improved Fuzzy C-Means yaitu dengan menyediakan lima variasi parameter pembobot (m).

2. Pengujian terhadap banyaknya data yang digunakan, yaitu menyediakan sebanyak lima kelompok data dan setiap kelompok data tersebut terdiri dari beberapa jumlah data yang berbeda-beda.
3. Pengujian membandingkan metode bobot Wtf dan Improved Fuzzy C-Means.

3.6 Kesimpulan dan Saran

Penarikan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi, dan pengujian sistem telah selesai dilakukan. Kesimpulan diambil dari hasil pengujian dan analisis terhadap sistem yang dibangun. Saran bertujuan untuk memperbaiki kesalahan atau kekurangan yang ada dan untuk pengembangan penelitian selanjutnya.

BAB IV
IMPLEMENTASI

Bab ini membahas implementasi perangkat lunak berdasarkan hasil yang telah diperoleh dari metode penelitian dan perancangan pada bab sebelumnya. Bab ini terdiri dari penjelasan tentang implementasi sistem, batasan-batasan dalam implementasi, implementasi algoritma, implementasi antarmuka, dan implementasi metode.

4.1 Spesifikasi Sistem

Hasil analisis kebutuhan dan perancangan perangkat lunak yang telah diuraikan pada bab sebelumnya menjadi acuan untuk diimplementasikan menjadi aplikasi yang dapat berfngsi sesuai dengan kebutuhan.

4.1.1 Spesifikasi Perangkat Keras

Perangkat keras komputer yang digunakan dalam implementasi aplikasi memiliki spesifikasi sebagai berikut:

1. Processor Intel(R) Core(TM) i3-2330M
2. RAM 4 GB
3. Harddisk 500 GB
4. VGA Nvidia Geforce 610M 2 GB

4.1.2 Spesifikasi Perangkat Lunak

Perangkat lunak sistem komputer yang digunakan dalam implementasi aplikasi antara lain yaitu:

1. Sistem Operasi Windows 8.1 Pro
2. NetBeans IDE 8.0

4.2 Batasan Implementasi

Batasan implementasi adalah batasan kemampuan fungsional sistem berdasarkan perancangan yang telah diuraikan pada bab sebelumnya. Batasan implementasi dilakukan agar penelitian ini memiliki ruang lingkup yang jelas dalam mengimplementasikan sistem. Beberapa batasan implementasi adalah sebagai berikut:

1. Analisis sentimen *review* aplikasi *mobile* dirancang dan dijalankan pada perangkat berbasis *dekstop*.

2. Metode penyelesaian masalah yang digunakan adalah Improved Fuzzy C-Means.
3. Data yang digunakan sebagai *input* merupakan *review* yang diambil dari Google Play Store berbentuk dokumen teks (.txt).
4. Data yang digunakan adalah dokumen berbahasa Indonesia.
5. *Output* berupa hasil analisis sentimen yaitu positif atau negatif.

4.3 Implementasi Algoritma

Aplikasi ini terdiri dari tiga proses utama yaitu *preprocessing*, pembobotan, dan *clustering*. *Input* dokumen teks akan dilakukan *preprocessing* untuk mendapatkan kata sedemikian hingga siap diolah di proses pembobotan. Pembobotan dilakukan menggunakan metode pembobotan *term frequency* (TF). Dan metode *clustering* yang digunakan adalah Improved Fuzzy C-Means. Daftar fungsi pada aplikasi analisis sentimen ini ditunjukkan pada Tabel 4.1.

Tabel 4.1 Daftar Fungsi Proses Analisis Sentimen

No	Proses	Fungsi	Keterangan
1	Preprocessing	ProsesToken()	Fungsi untuk menghilangkan karakter selain huruf dan mengubah huruf ke huruf kecil semua.
		ProsesNormalisasi()	Fungsi untuk mengubah kata tidak baku menjadi kata baku.
		ProsesFilter()	Fungsi untuk menghilangkan kata tidak penting.
		ProsesKataDasar()	Fungsi untuk mengubah semua kata ke bentuk kata dasarnya.
2	Pembobotan	ProsesBobot()	Fungsi untuk menghitung bobot *term* positif dan negatif menggunakan pembobotan *term frequency* (TF).
3	Improved Fuzzy C-Means	ClusterAwal()	Fungsi untuk mendapatkan pusat *cluster* awal
		HitungKeanggotaan()	Fungsi untuk menghitung derajat keanggotaan setiap dokumen
		HitungFungsiObyektif()	Fungsi untuk menghitung nilai fungsi obyektif
		ClusterBaru()	Fungsi untuk mendapatkan pusat *cluster* baru

4.3.1 Proses Preprocessing

Tahap ini terdiri dari tokenizing, normalisasi, filtering, dan stemming. Potongan kode program proses preprocessing ditunjukkan pada Kode Program 4.1.

```
//tokenizing
Tokenizing tok = new Tokenizing(namaFile);
listReview = tok.ProsesToken();

//normalisasi
Normalisasi norm = new Normalisasi(listReview,
"dokumen\\KataBaku.txt");
listReview = norm.prosesNormalisasi();

//filtering
Filtering filter = new Filtering(listReview,
"dokumen\\StopWordIN.txt");
listReview = filter.ProsesFilter();

//stemming
ArifinSetiono stem = new ArifinSetiono(listReview,
"dokumen\\KataDasar.txt");
listReview = stem.ProsesKataDasar();
```

Kode Program 4.1. Implementasi preprocessing

4.3.1.1 Proses Tokenizing

Fungsi tokenizing digunakan untuk menghilangkan karakter selain huruf pada dokumen review dan menjadikannya huruf kecil semua. Implementasi dari tokenizing ditunjukkan pada Kode Program 4.2.

```
public ArrayList<ArrayList<String>> ProsesToken(
ArrayList<String> rev){
   ArrayList<ArrayList<String>> listReview = new
   ArrayList<ArrayList<String>>();

   for(int i=0; i<rev.size(); i++){
      ArrayList<String> list = new ArrayList<String>();
      String strLower = rev.get(i).toLowerCase();

      StringTokenizer st = new StringTokenizer(strLower, "
      1234567890`~!@#$%^&*)(-
      _=+}{][\\|;:'\",<.>/?ðÿŽž˜šâ™¡˜¥…œ,‘ ");
      while(st.hasMoreTokens()){
         String str = st.nextToken();
         list.add(str);
      }
      listReview.add(list);
   }

   return listReview;
}
```

Kode Program 4.2. Implementasi tokenizing

4.3.1.2 Proses Normalisasi

Fungsi normalisasi digunakan untuk mengubah kata singkatan menjadi kata aslinya dan kata tidak baku menjadi kata baku. Kata yang akan dinormalisasi adalah kata yang tersedia di dalam kamus kata ternormalisasi seperti yang ditunjukkan pada Lampiran1 1. Implementasi fungsi normalisasi ditunjukkan pada Kode Program 4.3.

```
ArrayList<ArrayList<String>> listReview;
String namaFile;
public ArrayList<ArrayList<String>> prosesNormalisasi(){
   HashMap<String, String> kamus = new HashMap<String,
   String>();
   try {
      Scanner sc = new Scanner(new File(namaFile));
      while (sc.hasNextLine()) {
         String[] str = sc.nextLine().split("=");
         kamus.put(str[0], str[1]);
      }

      for(int i=0; i<listReview.size(); i++){
         for(int j=0; j<listReview.get(i).size(); j++){
            if(kamus.containsKey(listReview.get(i).get(j))){
               listReview.get(i).set(j,kamus.get(listReview
               .get(i).get(j)));
            }
         }
      }
      } catch (FileNotFoundException ex) {
            System.out.println(ex);
      }

      return listReview;
}
```

Kode Program 4.3. Implementasi normalisasi

4.3.1.3 Proses Filtering

Fungsi filtering bertujuan untuk menghilangkan kata yang tidak penting. Kata yang ada di dalam dokumen dan ada di dalam kamus stopword akan dihapus sehingga tersisa kata yang unik. Daftar kata stopword ditunjukkan pada Lampiran 2. Implementasi dari filtering ditunjukkan pada Kode Program 4.4.

```
ArrayList<ArrayList<String>> listReview;
String namaFile;
public ArrayList<ArrayList<String>> ProsesFilter(){
   HashSet<String> stoplist = new HashSet<String>();

   try {
      Scanner sc = new Scanner(new File(namaFile));
      while (sc.hasNextLine()) {
         String str = sc.nextLine();
```

```
            stoplist.add(str);
        }

        for(int i=0; i<listReview.size(); i++){
            for(int j=0; j<listReview.get(i).size(); j++){
                if(stoplist.contains(listReview.get(i).get(j))){
                    listReview.get(i).remove(j);
                    j-=1;
                }
            }
        }
        } catch (FileNotFoundException ex) {
            System.out.println(ex);
        }

        return listReview;
}
```

Kode Program 4.4. Implementasi filtering

4.3.1.4 Proses Stemming

Fungsi stemming digunakan untuk mengubah kata ke bentuk kata dasarnya. Kata yang akan diubah ke bentuk kata dasarnya adalah kata yang tersedia di dalam kamus kata dasar. Daftar kata dasar ditunjukkan pada Lampiran 3. Metode stemming yang digunakan adalah stemming Arifin – Setiono. Implementasi proses stemming ditunjukkan pada Kode Program 4.5.

```
ArrayList<ArrayList<String>> listReview;
String namaFile;
public ArrayList<ArrayList<String>> ProsesKataDasar(){
    String[] listAwalan = {"me","di","ke","pe","se","be","ku",
    "ber","ter","per","mem","pem","men","pen","meng","meny",
    "peng"};

    String[] listAkhiran = {"i","an","in","mu","ku","lah","kah",
    "pun","kan","nya"};

    HashSet<String> listKDasar = new HashSet<String>();

    try {
        Scanner sc = new Scanner(new File(namaFile));
        while (sc.hasNextLine()) {
            String str = sc.nextLine();
            listKDasar.add(str);
        }

        for(int i=0; i<listReview.size(); i++){
            for(int j=0; j<listReview.get(i).size(); j++){
                String KD = listReview.get(i).get(j);
                //System.out.println(KD);
                String tmpKD = KD;
                boolean KETEMU = false;
                String[] awalan = {"",""};
                String[] akhiran = {"","",""};
```

```
if(listKDasar.contains(KD))
   continue;
else{
   for(int a=0; a<2; a++){
      for(int b=2; b<=4; b++){
         if(KD.length()<5)
            continue;
         String tmpAwalan = KD.substring(0, b);
         if(Arrays.asList(listAwalan).
         contains(tmpAwalan)){
            awalan[a] = tmpAwalan;
            tmpKD = KD.substring(b, KD.length());
               if(listKDasar.contains(tmpKD)){
                  KETEMU = true;
                  break;
               }
            }
         }
         KD=tmpKD;
         if(a>0 && awalan[a]=="" && awalan[a-1]!="")
            awalan[a]=awalan[a-1];
         if(KETEMU)
            break;
      }

      if(!KETEMU){
         for(int a=0; a<3; a++){
            for(int b=1; b<4; b++){
               if(KD.length()<5)
                  continue;
               String tmpAkhiran =
               KD.substring(KD.length()-b);
               if(Arrays.asList(listAkhiran).
               contains(tmpAkhiran)){
                  akhiran[a] = tmpAkhiran;
                  tmpKD = KD.substring(0,
                  KD.length()-b);
                  if(listKDasar.contains(tmpKD)){
                     KETEMU = true;
                     break;
                  }
               }
            }
            KD=tmpKD;
            if(a==0 && akhiran[a+1]=="" &&
            akhiran[a+2]=="")
               akhiran[a+2]=akhiran[a+1]=akhiran[a];
            if(a==1 && akhiran[a]=="" && akhiran[a-
            1]!="")
               akhiran[a+1]=akhiran[a]=akhiran[a-1];
            else if(a==2 && akhiran[a]=="" &&
            akhiran[a-1]!=""){
               akhiran[a]=akhiran[a-1];
               akhiran[a-1]=akhiran[a-2];
            }

            if(KETEMU)
               break;
         }
```

```
                if(listKDasar.contains(awalan[0]+awalan[1]+KD))
                   KD = awalan[0]+awalan[1]+KD;
                else if(listKDasar.contains(awalan[0]+
                awalan[1]+KD+akhiran[2]))
                   KD = awalan[0]+awalan[1]+KD+akhiran[2];
                else if(listKDasar.contains(awalan[0]+awalan[1]+
                KD+akhiran[2]+akhiran[1]))
                   KD = awalan[0]+awalan[1]+KD+akhiran[2]+
                   akhiran[1];
                else if(listKDasar.contains(awalan[1]+KD))
                   KD = awalan[1]+KD;
                else if(listKDasar.contains(awalan[1]+
                KD+akhiran[2]))
                   KD = awalan[1]+KD+akhiran[2];
                else if(listKDasar.contains(awalan[1]+
                KD+akhiran[2]+akhiran[1]))
                   KD = awalan[1]+KD+akhiran[2]+akhiran[1];
                else {
                   String[] khusus =
                   {"k","t","s","p","t","k","m","n"};
                   for(int a=0; a<khusus.length; a++){
                      if(listKDasar.contains(khusus[a]+KD)){
                         KD = khusus[a]+KD;
                         break;
                      }
                   }
                }
             }
          }

          listReview.get(i).set(j, KD);
       }
    }


    } catch (FileNotFoundException ex) {
       System.out.println(ex);
    }

    return listReview;
}
```

Kode Program 4.5. Implementasi stemming

4.3.2 Proses Pembobotan

Proses pembobotan dilakukan menggunakan pembobotan term frequency. Setiap dokumen dihitung jumlah term positif dan term negatif berdasarkan kamus term positif dan kamus term negatif seperti pada Lampiran 4 dan lampiran 5. Sedangkan kata negasi yang digunakan dalam pembobotan ini antara lain: jika, kalau, tanpa, tidak, kurang, dan belum. Pembobotan bertujuan untuk mengetahui bobot setiap dokumen dan hasil dari pembobotan akan digunakan sebagai fitur pada

proses clustering. Implementasi pembobotan term frequency ditunjukkan pada Kode Program 4.6.

```
ArrayList<ArrayList<String>> listReview;
String filePos, fileNeg, fileLabel;
public ArrayList<ArrayList<Double>> prosesBobotTF(){
   HashSet<String> kamusPos = new HashSet<String>();
   HashSet<String> kamusNeg = new HashSet<String>();
   ArrayList<int[]> tf = new ArrayList<int[]>();
   //dokumen<bobot>
   ArrayList<Double[]> bobot = new ArrayList<Double[]>();
   ArrayList<ArrayList<Double>> bobotAkhir = new
   ArrayList<ArrayList<Double>>();

   try {
      Scanner sc1 = new Scanner(new File(filePos));
      while (sc1.hasNextLine()) {
         String str = sc1.nextLine();
         kamusPos.add(str);
      }

      Scanner sc2 = new Scanner(new File(fileNeg));
      while (sc2.hasNextLine()) {
         String str = sc2.nextLine();
         kamusNeg.add(str);
      }

      String[] kataNegasi =
      {"jika","kalau","tanpa","tidak","kurang","belum"};
      for(int i=0; i<listReview.size(); i++){
         //[0]=pos, [1]=neg
         int[] tmpTF = {0, 0};
         Double[] tmpBobot = {0d, 0d};
         bobot.add(tmpBobot);
         tf.add(tmpTF);

         for(int j=0; j<listReview.get(i).size(); j++){
            String term = listReview.get(i).get(j);

            //jika ada kata negasi
            if(Arrays.asList(kataNegasi).contains(term)){
               if(j<listReview.get(i).size()-1){
                  String tmp = listReview.get(i).get(j+1);
                     if(kamusPos.contains(tmp)){
                        tmpTF[1]+=1;
                        j+=1;
                     }
                     else if(kamusNeg.contains(tmp)){
                        tmpTF[0]+=1;
                        j+=1;
                     }
               }
            }
            else {
               if(kamusPos.contains(term)){
                  tmpTF[0]+=1;
               }
               else if(kamusNeg.contains(term)){
```

```
                    tmpTF[1]+=1;
                }
            }

        }
        tf.set(i, tmpTF);

        if(tmpTF[0]!=0)
            tmpBobot[0]=1+Math.log10(tmpTF[0]);
        if(tmpTF[1]!=0)
            tmpBobot[1]=1+Math.log10(tmpTF[1]);

        bobot.set(i, tmpBobot);
        }

        for(int i=0; i<bobot.get(0).length; i++){
            ArrayList<Double> tmp = new ArrayList<Double>();
            for(int j=0; j<bobot.size(); j++){
                tmp.add(bobot.get(j)[i]);
            }
            bobotAkhir.add(tmp);
        }

        Scanner sc3 = new Scanner(new File(fileLabel));
        while (sc3.hasNextLine()) {
            String str = sc3.nextLine();
            label.add(Integer.parseInt(str));
        }

    } catch (FileNotFoundException ex) {
        System.out.println(ex);
    }

    return bobotAkhir;
}
```

Kode Program 4.6. Implementasi pembobotan term frequency

4.3.3 Proses Clustering

Proses analisis sentimen menggunakan clustering Improved Fuzzy C-Means memiliki empat tahapan yaitu pemilihan pusat cluster awal, menghitung derajat keanggotaan, menghitung fungsi obyektif, dan menghitung pusat cluster baru. Tahapan tersebut dilakukan berulang-ulang hingga mendapatkan hasil clustering yang tetap.

4.3.3.1 Proses Pemilihan Pusat Cluster Awal

Pada proses ini, akan dipilih beberapa dokumen untuk menjadi pusat cluster awal. Banyaknya cluster ditentukan oleh algoritma, dan jumlah yang dihasilkan merupakan jumlah terbaik. Implementasi pemilihan pusat cluster awal ditunjukkan pada Kode Program 4.7.

```
public int[] ClusterAwal(double[][] jarak){
   ArrayList<Integer> cAwal = new ArrayList<Integer>();
   int threshold = 3;
   ArrayList<ArrayList<Double>> jrk = new
   ArrayList<ArrayList<Double>>();
   ArrayList<Integer> idDok = new ArrayList<Integer>();

   for(int x=0; x<jarak.length; x++){
      ArrayList<Double> tmp = new ArrayList<Double>();
      for(int y=0; y<jarak.length; y++){
         tmp.add(jarak[x][y]);
      }
      idDok.add(x);
      jrk.add(tmp);
   }

   int N = 0;

   do {
      N = jrk.size();

      //average distance (Di)
      double[] average = new double[N];
      for(int i=0; i<N; i++){
         for(int j=0; j<N; j++){
            average[i] += jrk.get(i).get(j);
         }
         average[i] /= N-1;
      }

      //banyak dokumen kurang dari Di (ti)
      int[] te_i = new int[N];
      for(int i=0; i<N; i++){
      int temp=0;
      for(int j=0; j<N; j++){
         if(jrk.get(i).get(j)<average[i])
            temp += 1;
         }
         te_i[i] = temp;
      }

      //jarak terbesar yang kurang dari average (Di_max)
      double[] di_max = new double[N];
      for(int i=0; i<N; i++){
         double temp=0;
         for(int j=0; j<N; j++){
            if(jrk.get(i).get(j)<average[i] &&
            jrk.get(i).get(j)>=temp)
               temp = jrk.get(i).get(j);
         }
         di_max[i] = temp;
      }

      //sample density (Pi)
      double[] pe_i = new double[N];
      for(int i=0; i<N; i++){
         if(di_max[i]==0)
            pe_i[i]=0;
         else
```

```
            pe_i[i] = te_i[i]/di_max[i];
        }

        //pilih dokumen sebagai cluster
        double maxpi=0;
        int awalCluster=0;
        for(int i=0; i<N; i++){
            if(pe_i[i]>maxpi){
                maxpi = pe_i[i];
                awalCluster=idDok.get(i);
            } else if(pe_i[i]==maxpi){
                awalCluster=idDok.get(i);
            }
        }

        cAwal.add(awalCluster);

        //seleksi dokumen yang dihapus
        ArrayList<Integer> seleksiDok = new ArrayList<Integer>();
        int index = idDok.indexOf(awalCluster);
        for(int i=0; i<N; i++){
            if(jrk.get(index).get(i)<=di_max[index]){
                seleksiDok.add(i);
            }
        }

        //hapus horisontal
        for(int i=0; i<seleksiDok.size(); i++){
            int j = seleksiDok.get(i)-i;
            jrk.remove(j);
            idDok.remove(j);
        }

        //hapus vertikal
        for(int i=0; i<jrk.size(); i++){
            for(int j=0; j<seleksiDok.size(); j++){
                int k = seleksiDok.get(j)-j;
                jrk.get(i).remove(k);
            }
        }

        N=jrk.size();
        } while (N>threshold);

        jmlCluster = cAwal.size();
        int[] c = new int[cAwal.size()];
        for(int i=0; i<cAwal.size(); i++)
            c[i] = cAwal.get(i);

        return c;
    }
```

Kode Program 4.7. Implementasi pemilihan pusat cluster awal

4.3.3.2 Proses Hitung Derajat Keanggotaan

Dalam proses ini akan dihitung derajat keanggotaan semua dokumen terhadap semua cluster. Nilai derajat keanggotaan yang diperoleh harus

dinormalisasi sedemikian hingga jumlah derajat keanggotaan setiap dokumen bernilai satu. Setiap dokumen memiliki kecenderungan menjadi anggota dari cluster dengan derajat keanggotaan terbesar. Proses hitung derajat keanggotaan diimplementasikan seperti pada Kode Program 4.8.

```
public double[][] HitungKeanggotaan(double[][] dt, double [][]
cluster){
   double[][] derajat = new
   double[dt[0].length][cluster[0].length];
   double[][] jrkDok, jrkCluster;

   Euclidean dist1 = new Euclidean(dt, cluster);
   jrkDok = dist1.getJarak();
   Euclidean dist2 = new Euclidean(cluster, cluster);
   jrkCluster = dist2.getJarak();

   //ij=dok dgn cluster, jk=cluster dgn cluster
   for(int i=0; i<jrkDok.length; i++){
      for(int j=0; j<jrkDok[i].length; j++){
         for(int k=0; k<jrkDok[i].length; k++){
            if(jrkDok[i][j]==0)
               derajat[i][j]+=1;
            if(j!=k)
               derajat[i][j]+=Math.pow(jrkDok[i][j]/
               jrkCluster[j][k], 2/(this.pembobot-1));
         }
      }
   }
   for(int i=0; i<jrkDok.length; i++){
      for(int j=0; j<jrkDok[i].length; j++){
         if(derajat[i][j]==jrkDok[i].length){
            for(int k=0; k<jrkDok[i].length; k++){
               if(k!=j)
                  derajat[i][k]=0;
               else
                  derajat[i][k]=1;
            }
            break;
         } else {
            if(derajat[i][j]!=0)
               derajat[i][j]=1/derajat[i][j];
         }
      }
   }

   return derajat;
}
```

Kode Program 4.8. Implementasi hitung derajat keanggotaan

4.3.3.3 Proses Hitung Fungsi Obyektif

Fungsi obyektif digunakan untuk mengetahui kualitas hasil clustering. Semakin kecil nilai fungsi obyektif semakin bagus kualiatas hasil clustering.

Implementasi dari penghitungan fungsi obyektif ditunjukkan pada Kode Program 4.9.

```
public double hitungFungsiObyektif(){
   double[][] aData = new double[jmlVariabel][jmlData];
   for(int x=0; x<jmlVariabel; x++){
      for(int y=0; y<jmlData; y++){
         aData[x][y] = data[x][y];
      }
   }

   double[][] aCluster = new double[jmlVariabel][jmlCluster];
   for(int x=0; x<jmlVariabel; x++){
      for(int y=0; y<jmlCluster; y++){
         aCluster[x][y] = pusatCluster[x][y];
      }

   }

   double[][] jarak;
   Euclidean dist = new Euclidean(aData, aCluster);
   jarak = dist.getJarak();

   double j = 0;
   for(int x=0; x<jarak.length; x++){
      for(int y=0; y<jarak[0].length; y++){
         double xx=0;
         xx = Math.pow(keanggotaan[x][y], pembobot) *
         Math.pow(jarak[x][y], 2);
         j += xx;
      }
   }

   return j;
}
```

Kode Program 4.9. Implementasi hitung fungsi obyektif

4.3.3.4 Proses Hitung Pusat Cluster Baru

Pusat cluster baru dihitung ketika selisih fungsi obyektif sekarang dengan sebelumnya masih lebih besar dari minimum toleransi (umumnya $1x10^{-5}$). Implementasi perhitungan pusat cluster baru ditunjukkan pada Kode Program 4.10.

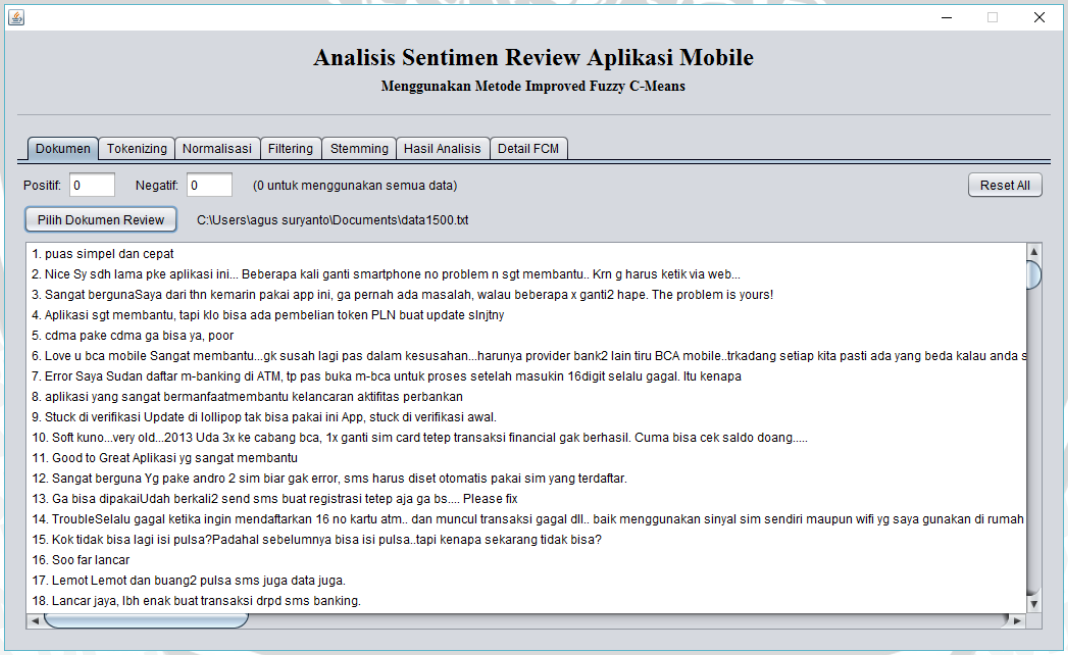
```
public double[][] ClusterBaru(){
   double[][] nCluster = new double[jmlVariabel][jmlCluster];
   double[] jmlKuadrat = new double[jmlCluster];

   for(int x=0; x<keanggotaan[0].length; x++){
      for(int y=0; y<keanggotaan.length; y++){
         jmlKuadrat[x] += Math.pow(keanggotaan[y][x], 2);
      }
   }

   for(int i=0; i<jmlCluster; i++){
      for(int j=0; j<jmlVariabel; j++){
```

```
            for(int k=0; k<jmlData; k++){
                nCluster[j][i] += data[j][k] *
                Math.pow(keanggotaan[k][i], 2);
            }
        }
    }

    for(int i=0; i<jmlVariabel; i++){
        for(int j=0; j<jmlCluster; j++){
            nCluster[i][j] /= jmlKuadrat[j];
        }
    }

    return nCluster;
}
```

Kode Program 4.10. Implementasi hitung pusat cluster baru

4.4 Implementasi Antarmuka

Dalam sub bab ini akan dijelaskan mengenai langkah-langkah yang dilakukan user untuk menjalankan aplikasi. Langkah pertama adalah user memilih dokumen yang akan dilakukan analisis sentimen, dalam hal ini dokumen *review* aplikasi mobile.

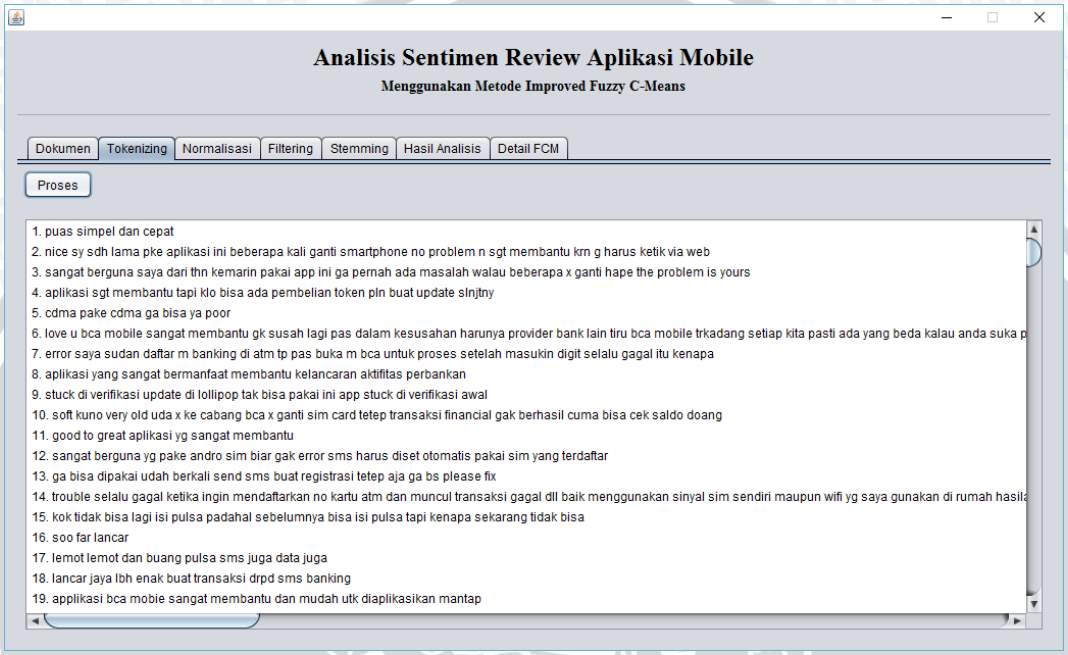


Gambar 4.1 Interface Input Dokumen

Dokumen yang menjadi input adalah dokumen dengan format .txt. Satu dokumen .txt ini berisi beberapa baris *review*, di mana satu baris *review* mewakili satu dokumen *review*. Pilih dokumen dengan cara klik tombol “Pilih Dokumen”, kemudian pilih dokumen .txt. Aplikasi akan menampilkan beberapa baris *review*

yang merupakan isi dari dokumen *review* seperti yang ditunjukkan pada Gambar 4.1.

Setelah yang sudah dipilih selanjutnya dilakukan proses *tokenizing* untuk menghilangkan karakter selain huruf dan menjadikan semua karakter ke huruf kecil semua dengan cara masuk ke tabulasi "Tokenizing" lalu klik tombol "Proses". Selanjutnya aplikasi akan menampilkan hasil seperti pada Gambar 4.2.

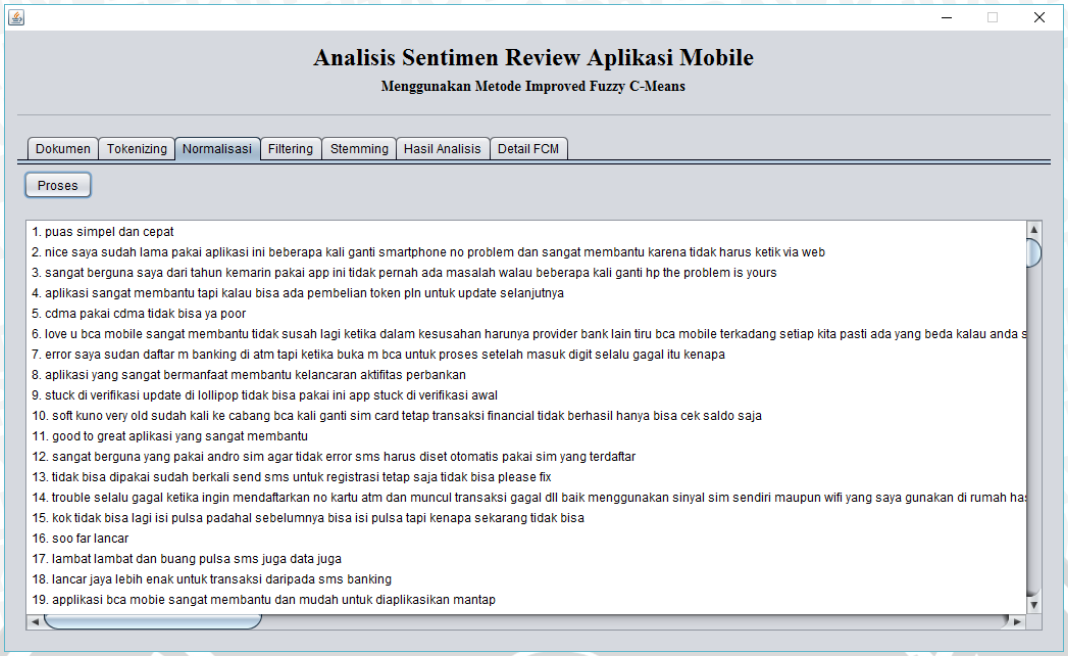


Gambar 4.2 Interface Tokenizing

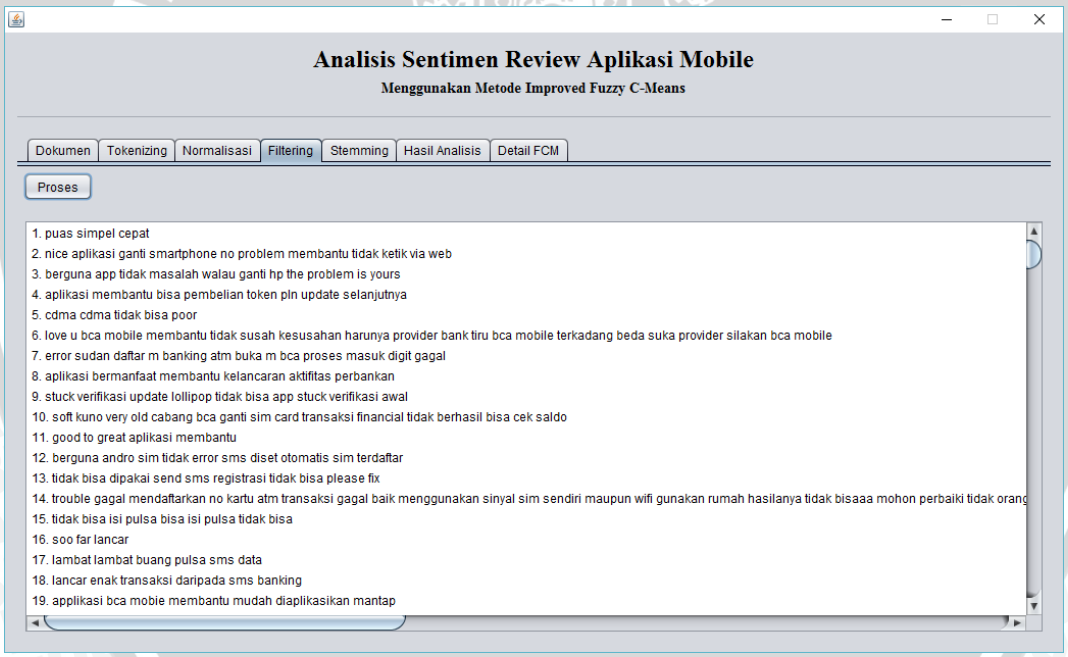
Seteleh dokumen di-*tokenizing*, langkah selanjutnya adalah melakukan normalisasi untuk mengubah kata yang disingkat ke bentuk aslinya dan mengubah kata tidak baku ke kata baku. Normalisasi dapat dilakukan dengan cara masuk ke tabulasi "Normalisasi" lalu klik tombol "Proses".

Langkah selanjutnya adalah *filtering* yang bertujuan untuk menghilangkan kata tidak penting, dilakukan dengan cara membuka tabulasi "Filtering" kemudian klik tombol "Proses". Sistem akan menampilkan hasil *filtering* seperti pada Gambar 4.4.

Setelah filtering, dokumen dilakukan stemming untuk mengubah semua kata ke bentuk kata dasarnya dengan cara menuju tabulasi "Stemming" lalu klik tombol "Proses". Sistem akan menampilkan hasil stemming seperti pada Gambar 4.5.

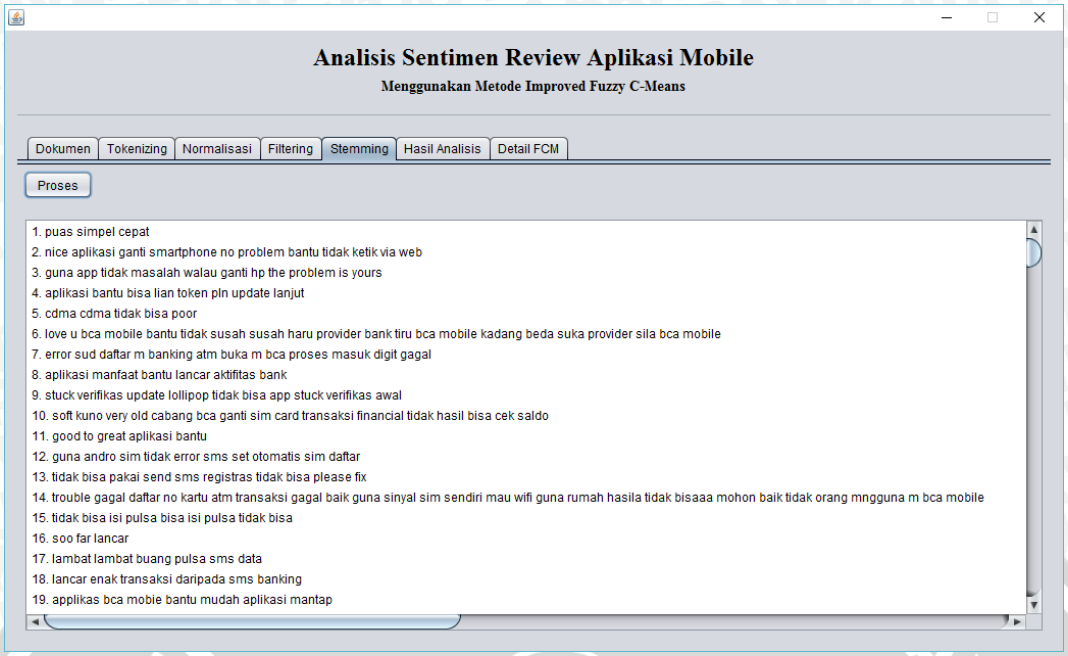


Gambar 4.3 Interface Normalisasi

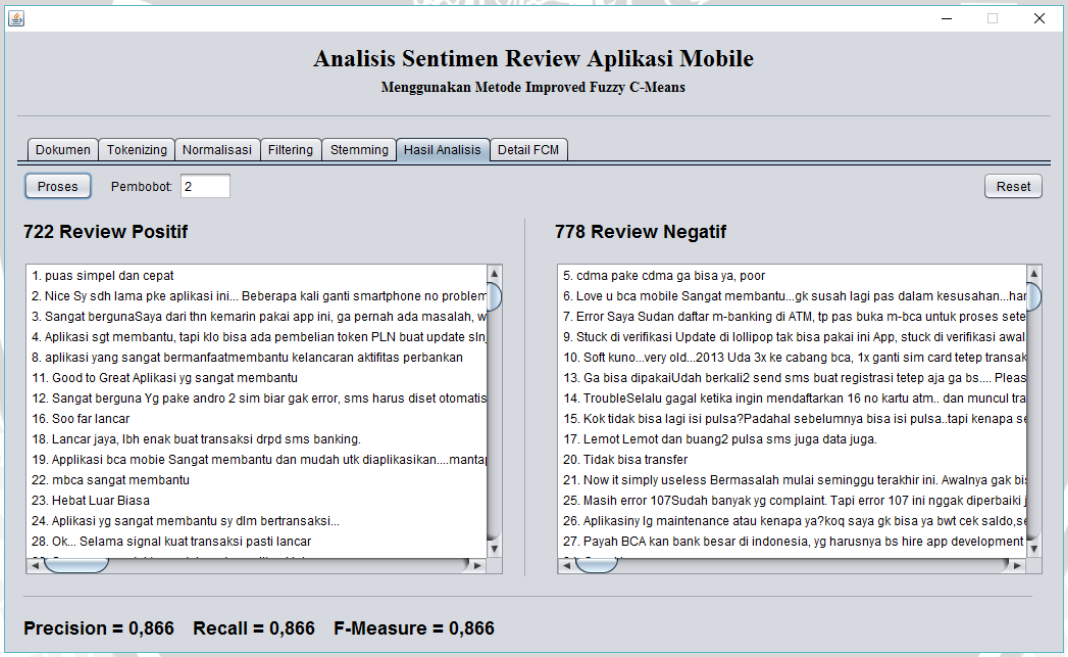


Gambar 4.4 Interface Filtering

Hasil analisis sentimen dapat dilihat pada tabulasi “Hasil Analisis” dengan klik tombol “Proses”. Sistem akan menampilkan hasil analisis sentimen seperti pada Gambar 4.6. Pada tabulasi ini juga menampilkan nilai rata-rata *precision*, *recall*, dan *f-measure* dari hasil analisis sentimen sistem.

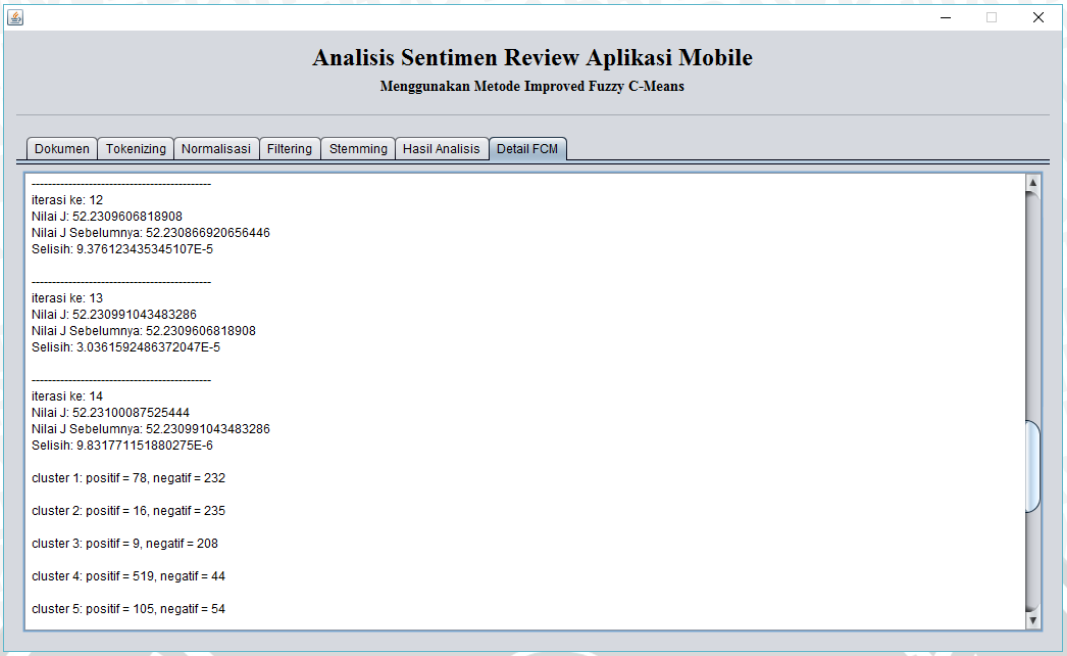


Gambar 4.5 Interface Stemming



Gambar 4.6 Interface Hasil Sentimen Analisis

Pada tabulasi "Detail FCM" seperti pada Gambar 4.7 menampilkan sebagian dari hasil proses perhitungan Improved Fuzzy C-Means diantaranya yaitu jumlah *cluster* yang terbentuk, nilai dan selisih fungsi obyektif dari setiap iterasi, jumlah dokumen di setiap *cluster*, nilai *precision*, *recall*, dan *f-measure* kelas positif dan negatif, dan rata-ratanya keduanya.



Gambar 4.7 Hasil Proses Improved FCM

BAB V
PENGUJIAN DAN ANALISIS

5.1 Pengujian

Bab ini membahas pengujian hasil analisis sentimen oleh sistem terhadap analisis sentimen oleh manusia. Berbagai macam pengujian dilakukan untuk mengetahui kondisi terbaik sistem terhadap berbagai macam permasalahan. Ada tiga pengujian yang dilakukan yaitu pengujian terhadap variasi jumlah dokumen positif dan dokumen negatif, pengujian terhadap parameter pembobot yang beragam, dan pengujian membandingkan metode Improved Fuzzy C-Means dengan klasifikasi menggunakan bobot Wtf.

5.2 Hasil dan Analisis Pengujian

Pengujian dilakukan terhadap 1500 dokumen review aplikasi BCA Mobile. Dokumen tersebut sebelumnya telah memiliki label yaitu positif atau negatif yang nantinya akan digunakan untuk mengukur nilai *precision*, *recall*, dan *f-measure*. Dalam pengujian ini dilakukan inisialisasi parameter Improved Fuzzy C-Means yaitu maksimum iterasi 100 dan minimum toleransi 0,00001.

5.2.1 Pengujian terhadap Variasi Jumlah Dokumen

Pengujian ini dilakukan untuk mengetahui kemampuan sistem terhadap jumlah dokumen yang tidak seimbang antara dokumen berlabel positif dan dokumen berlabel negatif. Pengujian dilakukan sebanyak tiga kali terhadap 750 dokumen yang dipilih secara acak dan diambil nilai rata-ratanya. Nilai precision, recall, dan f-measure pada pengujian dengan rasio jumlah dokumen positif dan negatif yang berbeda di setiap skenarionya ditunjukkan pada Tabel 5.1.

Tabel 5.1 Hasil Pengujian Variasi Jumlah Dokumen

Rasio (%)	Jumlah Dokumen			Precision	Recall	F-measure
	Positif	Negatif				
10:90	75	675	Random 1	77,12	83,04	79,66
			Random 2	78,15	82,07	79,93
			Random 3	78,85	87,04	82,21
			Rata-rata	78,04	84,05	80,6
30:70	225	525	Random 1	85,02	81,65	83,04
			Random 2	86,66	82,32	84,03
			Random 3	86,9	83,81	85,12
			Rata-rata	86,19	82,59	84,06
50:50	375	375	Random 1	85,08	85,07	85,07

			Random 2	86,27	86,27	86,27
			Random 3	86,4	86,4	86,4
			Rata-rata	85,92	85,91	85,91
70:30	525	225	Random 1	84,19	88,1	85,52
			Random 2	83,14	86,41	84,34
			Random 3	88,12	79,9	82,58
			Rata-rata	85,15	84,8	84,15
90:10	675	75	Random 1	79,55	75,63	77,4
			Random 2	80,41	78,22	79,26
			Random 3	80,72	78,89	79,77
			Rata-rata	80,23	77,58	78,81

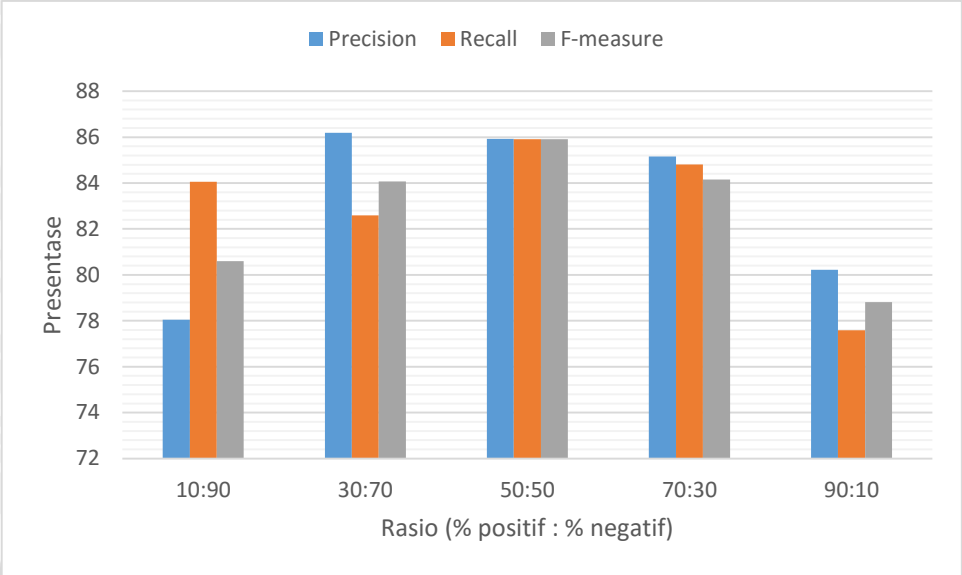
Berdasarkan Tabel 5.1 dapat dibuat tabel rata-rata nilai precision, recall, dan f-measure dari pengujian 1 seperti yang ditunjukkan Tabel 5.2. Hasil pengujian 1 dapat digambarkan dalam bentuk grafik seperti yang ditunjukkan pada Gambar 5.1.

Tabel 5.2 Rata-Rata Precision, Recall, F-measure Pengujian 1

Rasio (%)	**Jumlah Dokumen**		**Precision**	**Recall**	**F-measure**
	Positif	**Negatif**			
10:90	75	675	78,04	84,05	80,6
30:70	225	525	86,19	82,59	84,06
50:50	375	375	85,92	85,91	85,91
70:30	525	225	85,15	84,8	84,15
90:10	675	75	80,23	77,58	78,81

Pengujian 1 dari Tabel 5.2 menunjukkan bahwa *f-measure* tertinggi didapatkan ketika menggunakan dokumen dengan proporsi yang sama antara dokumen berlabel positif dengan dokumen berlabel negatif yaitu 50% berbanding 50%. Dan *f-measure* terendah didapatkan ketika menggunakan dokumen dengan proporsi 90% dokumen positif dan 10% dokumen negatif.

Semakin seimbang proporsi dokumen yang digunakan (50:50), semakin semakin tinggi nilai *f-measure* yang didapatkan. Sebaliknya, semakin besar perbedaan proporsi dokumen yang digunakan baik dokumen berlabel positif yang memiliki proporsi lebih tinggi (70:30, 90:10) atau dokumen berlabel negatif yang memiliki proporsi lebih tinggi (10:90, 30:70), nilai *f-measure* yang didapatkan semakin kecil.



Gambar 5.1 Grafik Rata-Rata *Precision*, *Recall*, dan *F-measure* Pengujian 1

Semakin seimbang proporsi antara dokumen berlabel positif dengan dokumen berlabel negatif, hasil clustering yang dihasilkan akan semakin baik. Ketika beberapa dokumen dalam jumlah yang besar memiliki karakteristik yang sama, dokumen tersebut akan lebih mudah untuk membentuk satu *cluster*. Saat dokumen yang memiliki *term* positif lebih banyak berada disekitar dokumen yang memiliki *term* positif lebih banyak juga, maka akan terbentuk satu *cluster* yang terdiri dari dokumen berlabel positif. Begitu juga saat dokumen yang memiliki *term* negatif lebih banyak berada disekitar dokumen yang memiliki *term* negatif lebih banyak juga, maka akan terbentuk satu *cluster* yang terdiri dari dokumen berlabel negatif.

Jumlah dokumen yang tidak seimbang akan menghasilkan analisis sentimen yang kurang baik. Analisis sentimen cenderung mengelompok ke kelas yang memiliki proporsi dokumen yang lebih besar. Semakin sedikit dokumen dalam suatu kelompok, tingkat kesamaan antar dokumen tersebut semakin rendah. Hal tersebut menyebabkan peluang suatu dokumen menjadi anggota dari kelas lain semakin tinggi. Sehingga tidak sedikit dokumen yang salah masuk ke *cluster* yang bukan seharusnya.

5.2.2 Pengujian terhadap Variasi Nilai Pembobot

Pengujian ini dilakukan untuk mengetahui nilai pembobot (m) yang terbaik untuk menghasilkan sistem yang efektif berdasarkan nilai f-measure. Nilai

pembobot yang digunakan adalah m>1 antara lain 1,1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 15; 20; 25; 50; dan 75. Rasio jumlah dokumen terbaik pada pengujian sebelumnya akan digunakan dalam pengujian ini yaitu 50% dokumen berlabel positif dan 50% dokumen berlabel negatif. Pengujian dilakukan sebanyak tiga kali terhadap 1300 dokumen yang dipilih secara acak, terdiri dari 650 dokumen berlabel positif dan 650 dokumen berlabel negatif kemudian diambil nilai rata-ratanya. Hasil pengujian variasi pembobot ditunjukkan pada Tabel 5.3.

Tabel 5.3 Hasil Pengujian Variasi Nilai Pembobot

Pembobot (m)		Precision	Recall	F-measure
1,1	Random 1	86,69	86,69	86,69
	Random 2	86,7	86,69	86,69
	Random 3	86,55	86,54	86,53
	Rata-rata	86,64	86,64	86,63
2	Random 1	86,77	86,77	86,77
	Random 2	86,72	86,69	86,69
	Random 3	86,85	86,85	86,85
	Rata-rata	86,78	86,77	86,77
3	Random 1	86,47	86,46	86,54
	Random 2	86,62	86,62	86,61
	Random 3	86,69	86,69	86,69
	Rata-rata	86,59	86,59	86,61
4	Random 1	86,71	86,69	86,69
	Random 2	86,62	86,62	86,62
	Random 3	86,62	86,62	86,62
	Rata-rata	86,65	86,64	86,64
5	Random 1	86,56	86,54	86,54
	Random 2	86,5	86,46	86,46
	Random 3	86,54	86,54	86,54
	Rata-rata	86,53	86,51	86,51
6	Random 1	86,79	86,77	86,77
	Random 2	86,55	86,54	86,54
	Random 3	86,7	86,7	86,7
	Rata-rata	86,68	86,67	86,67
7	Random 1	86,86	86,85	86,85
	Random 2	86,85	86,85	86,86
	Random 3	86,69	86,69	86,69
	Rata-rata	86,8	86,79	86,8
8	Random 1	86,55	86,54	86,54
	Random 2	86,7	86,69	86,69
	Random 3	86,71	86,69	86,69
	Rata-rata	86,65	86,64	86,64
9	Random 1	86,86	86,85	86,85
	Random 2	86,78	86,77	86,77

	Random 3	86,86	86,85	86,85
	Rata-rata	86,83	86,82	86,82
10	**Random 1**	86,63	86,62	86,61
	Random 2	86,93	86,92	86,92
	Random 3	86,8	86,77	86,77
	Rata-rata	86,79	86,77	86,77
15	**Random 1**	86,62	86,62	86,61
	Random 2	86,63	86,62	86,61
	Random 3	86,7	86,69	86,69
	Rata-rata	86,65	86,64	86,64
20	**Random 1**	86,62	86,62	86,62
	Random 2	86,71	86,69	86,69
	Random 3	86,7	86,69	86,69
	Rata-rata	86,68	86,67	86,67
25	**Random 1**	86,85	86,85	86,85
	Random 2	86,54	86,54	86,54
	Random 3	86,78	86,77	86,77
	Rata-rata	86,72	86,72	86,72
50	**Random 1**	86,47	86,46	86,46
	Random 2	86,78	86,77	86,77
	Random 3	86,78	86,77	86,77
	Rata-rata	86,68	86,67	86,67
75	**Random 1**	86,77	86,77	86,77
	Random 2	86,71	86,69	86,69
	Random 3	86,62	86,62	86,62
	Rata-rata	86,7	86,69	86,69

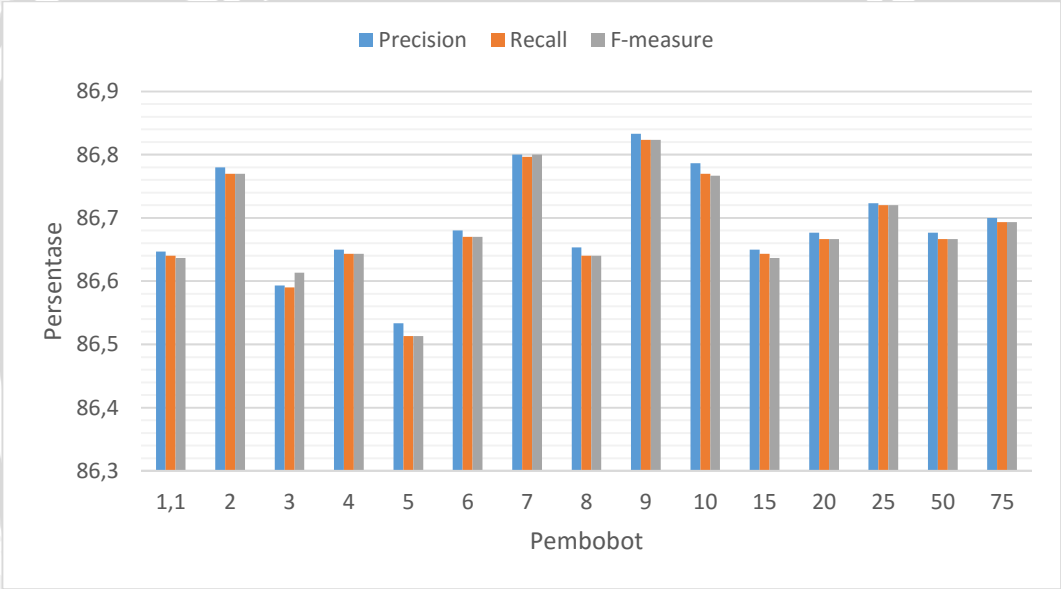
Berdasarkan Tabel 5.3 dapat dibuat tabel rata-rata nilai precision, recall, dan f-measure pengujian 2 seperti yang ditunjukkan Tabel 5.4. Hasil pengujian 2 dapat gambarkan dalam bentuk grafik seperti pada Gambar 5.2.

Tabel 5.4 Rata-Rata Precision, Recall, F-measure Pengujian 2

Pembobot (m)	Precision	Recall	F-measure
1,1	86,64	86,64	86,63
2	86,78	86,77	86,77
3	86,59	86,59	86,61
4	86,65	86,64	86,64
5	86,53	86,51	86,51
6	86,68	86,67	86,67
7	86,8	86,79	86,8
8	86,65	86,64	86,64
9	86,83	86,82	86,82
10	86,79	86,77	86,77
15	86,65	86,64	86,64
20	86,68	86,67	86,67
25	86,72	86,72	86,72

50	86,68	86,67	86,67
75	86,7	86,69	86,69

Pengujian 2 berdasarkan Tabel 5.2 menunjukkan bahwa rata-rata f-measure tertinggi didapatkan ketika menggunakan nilai pembobot m=9 yaitu 86,82%. Sedangkan rata-rata f-measure terendah didapatkan ketika menggunakan nilai pembobot m=5 yaitu 86,51%. Seiring dengan bertambahnya nilai pembobot yang digunakan, rata-rata f-measure yang didapatkan tidak mengalami peningkatan atau penurunan yang konstan. Sistem menghasilkan rata-rata f-measure yang naik turun dan tidak beraturan. Semua nilai pembobot yang digunakan menghasilkan rata-rata f-measure yang berada di kisaran lebih dari 86% dan kurang dari 87%.



Gambar 5.2 Grafik Rata-Rata Precision, Recall, dan F-measure Pengujian 2

Berdasarkan Persamaan (2-3), nilai pembobot berpengaruh terhadap nilai derajat keanggotaan. Semakin kecil nilai pembobot, nilai derajat keanggotaan suatu dokumen terhadap suatu *cluster* akan semakin mendekati 0 atau 1 seperti yang terjadi pada *hard clustering.* Sebaliknya semakin besar nilai pembobot, nilai derajat keanggotaan suatu dokumen terhadap suatu *cluster* tidak jauh berbeda dengan nilai derajat keanggotaan dokumen tersebut terhadap *cluster* yang lainnya. Walaupun mempengaruhi nilai derajat keanggotaan, nilai pembobot yang digunakan tidak berpengaruh terhadap hasil analisis sentimen karena hasil clustering ditentukan dari perbedaan nilai derajat keanggotaan. Perbedaan nilai derajat keanggotaan, baik

besar maupun kecil, suatu dokumen tetap memiliki kecenderungan masuk ke dalam satu cluster tertentu.

Berdasarkan Persamaan (2-6), nilai pembobot berpengaruh terhadap nilai fungsi obyektif. Semakin kecil nilai pembobot, fungsi obyektif akan semakin besar. Dengan semakin besarnya fungsi obyektif, sistem akan semakin lama mencapai minimum toleransi dan kesempatan untuk memperbaiki pusat *cluster* semakin besar. Sebaliknya semakin besar nilai pembobot, fungsi obyektif akan semakin kecil. Fungsi obyektif yang kecil menyebabkan sistem akan semakin cepat mencapai minimum toleransi dan kesempatan untuk memperbaiki pusat *cluster* semakin kecil.

5.2.3 Pengujian Metode Bobot Wtf dan Improved Fuzzy C-Means

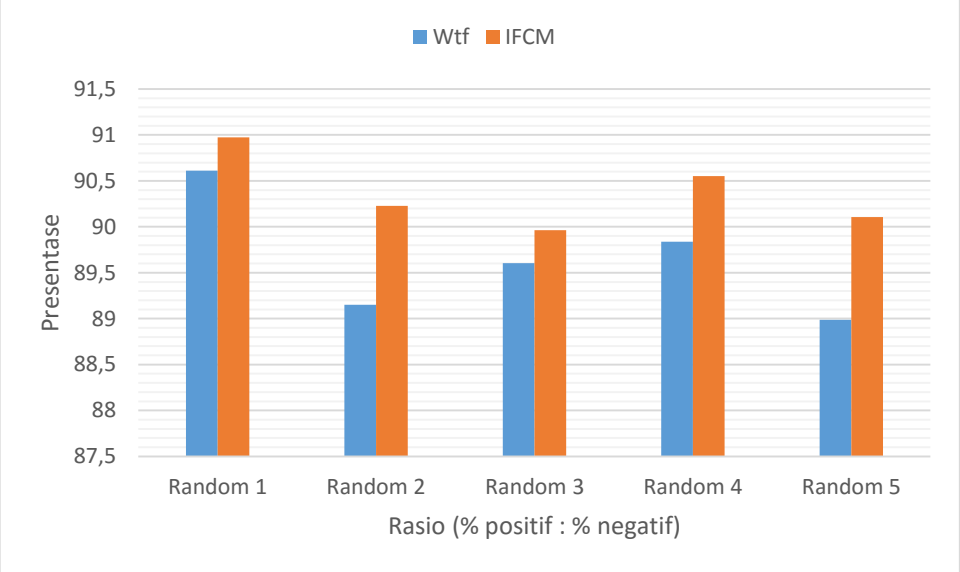
Pengujian ini dilakukan untuk mengetahui seberapa baik metode *Improved* FCM dibandingkan dengan menggunaan bobot Wtf sebagai metode klasifikasi. Bobot Wtf yang digunakan sebagai metode klasifikasi dilakukan dengan cara membandingkan bobot positif dan bobot negatif dalam suatu dokumen. Jika suatu dokumen memiliki bobot positif lebih besar, maka dokumen tersebut akan dimasukkan ke dalam kelas positif. Sebaliknya jika suatu dokumen memiliki bobot negatif lebih besar, maka dokumen tersebut akan dimasukkan ke dalam kelas negatif. Dokumen yang memiliki bobot positif sama dengan bobot negatif, tidak akan digunakan dalam pengujian. Pengujian dilakuakan sebanyak lima kali terhadap data acak yang terdiri dari 500 dokumen berlabel positif dan 500 dokumen berlabel negatif. Hasil pengujian penggunaan metode pembobotan Wtf dan *Improved* FCM ditunjukkan pada tabel 5.5. Nilai f-measure pada pengujian 3 dapat digambarkan dalam bentuk grafik seperti pada Gambar 5.3.

Tabel 5.5 Rata-Rata Precision, Recall, F-measure Pengujian 3

	Precision		**Recall**		**F-measure**	
	Wtf	**IFCM**	**Wtf**	**IFCM**	**Wtf**	**IFCM**
Random 1	90,817	91,186	90,634	90,974	90,612	90,974
Random 2	89,476	90,271	89,283	90,205	89,15	90,227
Random 3	89,786	90,131	89,63	89,96	89,605	89,964
Random 4	90,067	90,66	89,886	90,536	89,836	90,551
Random 5	89,353	90,195	89,005	90,113	88,986	90,105

Pengujian 3 berdasarkan Tabel 5.5 didapatkan bahwa nilai f-measure terendah sebesar 88,986% pada Random 5 dengan menggunakan Wtf dan f-

measure tertinggi sebesar 90,974% pada Random 1 dengan menggunakan *Improved* FCM. Dari Random 1 hingga Random 5, f-measure dari Improved FCM selalu lebih tinggi dibandingkan dengan f-measure dari Wtf, tetapi perbedaan nilainya tidak besar yaitu kurang dari 2%.



Gambar 5.3 Grafik F-Measure Bobot Wtf dan Improved Fuzzy C-Means

Penggunaan *Improved* FCM selalu menghasilkan f-measure yang lebih tinggi dari Wtf tetapi perbedaan nilainya tidak besar yaitu kurang dari 2%. Hal ini menunjukkan bahwa *Improved* FCM tidak memiliki kontribusi yang cukup besar terhadap analisis sentimen karena tanpa menggunakan *Improved* FCM, yaitu menggunakan bobot Wtf cukup untuk menghasilkan analisis sentimen yang baik.

Pada pengujian 3 ini, dokumen yang digunakan hanya dokumen yang memiliki bobot positif yang berbeda dengan bobot negatif, dokumen yang memiliki bobot yang sama tidak digunakan. *Improved* FCM tidak memiliki kontribusi yang cukup besar ketika dokumen yang digunakan memiliki perbedaan bobot positif dan negatif, sehingga bobot Wtf cukup untuk analisis sentimen. Tetapi *Improved* FCM memiliki kontribusi yang besar terhadap dokumen yang memiliki bobot positif yang sama dengan bobot negatif. Pada pengujian ini, nilai f-measure tertinggi sebesar 90,97% dengan dokumen yang digunakan memiliki bobot yang berbeda. Dan pada pengujian sebelumnya, nilai f-measure tertinggi sebesar 86,82% dengan dokumen yang digunakan ada yang memiliki bobot yang sama dan juga ada yang berbeda. Jika nilai f-measure pada pengujian sebelumnya merupakan nilai rata-rata

antara dokumen yang memiliki bobot yang sama dan dokumen yang memiliki bobot yang berbeda, maka didapatkan nilai f-measure untuk dokumen dengan bobot yang sama adalah sebesar 82,86%. Angka tersebut menunjukkan bahwa dalam analisis sentimen yang menggunakan dua fitur yaitu bobot posisit dan bobot negatif, *Improved* FCM mampu menangani dengan baik dokumen yang memiliki bobot yang sama.

BAB VI
KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil pengujian dan pembahasan yang telah dilakukan dapat diambil kesimpulan sebagai berikut:

1. Improved Fuzzy C-means mampu secara otomatis menemukan jumlah cluster terbaik yang bisa dibentuk dan menemukan pusat cluster awal mendekati pusat cluster yang sebenarnya. Dengan begitu, sistem akan dengan cepat mendapatkan hasil clustering yang konvergen.
2. Semakin seimbang jumlah antara dokumen positif dengan dokumen negatif yang digunakan, semakin baik hasil clustering yang didapatkan. Nilai f-measure tertinggi yang dihasilkan sebesar 86,82% dengan rasio jumlah dokumen yang seimbang antara dokumen positif dengan dokumen negatif. Dan nilai f-measure terendah sebesar 78,81% dengan rasio jumlah dokumen yang tidak seimbang.
3. Besar kecilnya nilai pembobot tidak berpengaruh terhadap hasil clustering, karena hasil clustering didapatkan dari perbedaan nilai derajat keanggotaan. Nilai pembobot berpengaruh terhadap kecepatan pemrosesan sistem hingga mencapai minimum toleransi. Semakin besar nilai pembobot, semakin cepat sistem melakukan pemrosesan dan kesempatan sistem memperbaiki pusat cluster semakin kecil. Sebaliknya semakin kecil nilai pembobot, semakin lama sistem melakukan pemrosesan dan kesempatan sistem memperbaiki pusat cluster semakin besar.

6.2 Saran

Untuk penelittian lebih lanjut disarankan memperbaiki algoritma stemming karena banyak term positif dan term negatif yang berimbuhan dalam penggunaannya di kehidupan sehari-hari. Selain itu diharapkan sistem dapat mengenali makna kalimat secara utuh terutama dalam penggunaan frase yang lebih dari dua kata. Agar pelabelan data lebih akurat, disarankan untuk dilakukan oleh orang yang memang ahli dalam menganalisis sentimen.

DAFTAR PUSTAKA

[ARI-02] Arifin, A., Setiono, A., 2002, "Klasifikasi Dokumen Berita Kejadian Berbahasa Indonesia dengan Algoritma Single Pass Clustering", Institut Teknologi Sepuluh November (ITS).

[BRA-13] Brandt, M. 2013. Google overtakes Apple. http://www.statista.com/chart/812/number-of-apps-available-in-the-top-app-stores/ [27 Nov 2014].

[CHA-05] Chaovalit, P., Zhou, L., 2005, "Moview review mining: A comparison between supervised and unsupervised classification approaches", in Proceedings of the Hawaii International Conference on System Sciences.

[DEN-10] Deng, J., Hu, J., Chi, H., Wu, J., 2010, "An Improved Fuzzy Clustering Method for Text Mining", Second International Conference on Network Security, Wireless Communications and Trusted Computing.

[FEL-07] Feldman, R., & Sanger, J. 2007. *The Text Mining Handbook*. New York: Cambridge University Press.

[GOO-09] Go, A., Huang, L., Bhayani, R., 2009, "Twitter Sentiment Analysis", Finnal Project from CS224N for Spring 2008/2009, The Stanford Natural Language Processing Group.

[KUS-09] Kusrini. Luthfi, E. 2009. *Algoritma Data Mining*. Jakarta: Penerbit Andi.

[LIU-10] Liu, B. 2010. "Sentiment Analysis and Subjectivity", handbook of naural language processing 2: 627-666.

[LIU-12] Liu, B. 2012. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.

[MAN-09] Manning, C. D., Raghavan, P., & Schutze, H. 2009. *An Introduction to Information Retrieval*. Cambridge: Cambridge University Press.

[MUS-09] Mustafa, Atika, Ali Akbar, and Ahmer Sultan. 2009, "Knowledge Discovery using Teks Mining: A Programmable Implementation

on Information Extraction and Categorization", international Journal of Multimedia and Ubiquitous Engineering Vol. 4, No. 2, April.

[MAU-13] Maulana, A. 2013. Akan ada 103,7 juta pengguna *smartphone* di Indonesia. http://tekno.liputan6.com/read/731892/akan-ada-1037-juta-pengguna-smartphone-di-indonesia [27 Nov 2014].

[NER-12] Neri, F., Aliprandi, C., Capeci, F., Cuadros, M., By, T., 2012, "Sentiment Analysis on Social Media", International Conference on Advances in Social Network Analysis and Mining.

[PAN-08] Pang, B., Lee, L., 2008, "Opinion Mining and Sentiment Analysis", in Foundations and Trends in Information Retrieval, vol. 2, no. Issue 1-2, pp. 1-135.

[SHA-12] Shanthi R., Suganya R., 2012, "Fuzzy C-Means Algorithm-A Review", International Journal of Scientifics and Research Publication, Vol. 2, Issue 11.

[WAN-10] Wang, J., Wang, S. T., 2010, "Double indices FCM algorithm based on hybrid distance metric learning", Journal of Software,Vol. 21, No. 8, pp.1878-1888.

[WIC-13] Wicaksono, Farizki, A., Nio, E., Myaeng, S., 2013, "Unsupervised Aprroach for Sentiment Analysis on Indonesian Movie Reviews", Dept. of Computer Science, Korea Advanced Institute of Science and Technology Daejon, Rep. of Korea.

[YAN-13] Yang, H., Han, D., Yu, F., 2013, "Improved Fuzzy C-Means Clustering Algorithm based on Sample Density", Journal of Theoretical and Applied Information Technology.

Lampiran 1 Kamus Normalisasi

Tabel 1 Kamus Normalisasi

Kamus Normalisasi		
abal=jelek	ginian=beginian	ngurusin=mengurusi
abis=habis	gitu=begitu	ngga=tidak
ad=ada	gj=jelek	nggak=tidak
aj=saja	gk=tidak	ngudek=mengurusi
aja=saja	gmana=bagaimana	nh=nih
ajah=saja	gmna=bagaimana	nich=nih
aje=saja	gnti=ganti	nie=ini
ajib=bagus	gregetan=emosi	niy=nih
aktifin=aktif	gt=begitu	njawab=menjawab
alias=atau	gtu=begitu	nmor=nomor
aliaz=atau	gue=saya	nmr=nomor
ama=dengan	gw=saya	nomer=nomor
ampe=sampai	hape=hp	np=kenapa
ampir=hampir	hars=harus	npa=kenapa
ancur=hancur	hbs=habis	nsbh=nasabah
ane=saya	hr=hari	numpuk=menumpuk
ap=apa	hri=hari	nunggu=tunggu
apaan=apa	hrus=harus	ny=nya
apik=bagus	hrs=harus	nyalahin=menyalahkan
aplgi=apalagi	hrsnya=harus	nyampah=sampah
aplikasiny=aplikasinya	ilang=hilang	nyampe=sampai
aplkasiny=aplikasinya	jd=jadi	nyebelin=menyebalkan
apus=hapus	jg=juga	nyedot=menyedot
ari=dari	jga=juga	nyerah=menyerah
ato=atau	jgn=jangan	nyesel=menyesal
atow=atau	jik=jika	nyiksa=menyiksa
atw=atau	jmn=jaman	nyoba=coba
aq=saya	jng=jangan	nyusahin=susah
baguss=bagus	joz=jos	org=orang
bahlul=bodoh	k=ke	pake=pakai
banget=sekali	kacangan=jelek	pakw=pakai
bangt=sekali	kaga=tidak	pas=ketika
bberapa=beberapa	kagak=tidak	pastiin=pastikan
bbrp=beberapa	kale=kali	payah=buruk
bedasarkan=berdasarkan	kalo=kalau	perna=pernah
beginian=begini	kalu=kalau	pd=pada
bejibun=banyak	karna=karena	pdhal=padahal
bener=benar	kasi=kasih	pdhl=padahal
beneren=benar	katro=lama	pngguna=pengguna

benerin=benar	kayak=seperti	persh=perusahaan
benerinnya=bener	kdng=kadang	perush=perusahaan
beneran=benar	kelar=selesai	pgn=ingin
broperasi=beroperasi	kena=terkena	pk=pakai
bgini=begini	kenceng=kencang	pke=pakai
bgitu=begitu	kepake=kepakai	pket=paket
bgt=sekali	kere=miskin	pny=punya
bgs=bagus	kesel=kesal	prusahaan=perusahaan
bgus=bagus	kirain=kira	psti=pasti
biar=agar	kl=kalau	rek=rekening
bikin=buat	klo=kalau	ribe=repot
bingit=sekali	klu=kalau	ribet=repot
bkin=buat	kluar=keluar	rmh=rumah
bli=beli	kmren=kemarin	sa=bisa
blm=belum	kmrn=kemarin	sableng=sinting
bln=bulan	kmudian=kemudian	sampe=sampai
blom=belum	kn=kan	sana=kesana
blgnya=bilangnya	knapa=kenapa	sbg=sebagai
bngt=sekali	knp=kenapa	sbgi=sebagai
bnr=benar	knpa=kenapa	sblmny=sebelumnya
bnyak=banyak	ko=kok	sblmnya=sebelumnya
bnyk=banyak	komplit=lengkap	sbtulnya=benar
brekele=parah	koq=kok	sdh=sudah
br=baru	kpn=kapan	sdkt=sedikit
brapa=berapa	krn=karena	sebenernya=benar
brhasil=berhasil	krna=karena	segede=sebesar
brkali=berkali	ktny=katanya	selese=selesai
bejal=berjalan	kudu=harus	seneng=senang
brp=berapa	kq=kok	sgera=segera
bru=baru	ky=seperti	sgt=sangat
bs=bisa	kyk=seperti	sgala=segala
bsa=bisa	lbh=lebih	sgl=segala
bsar=besar	lemot=lambat	sgra=segera
buat=untuk	lg=lagi	sich=sih
bukanya=bukannya	lgi=lagi	silahkan=silakan
bulak=bolak	lgsg=langsung	skali=sekali
bulukan=berbulu	liat=lihat	skli=sekali
bw=bawa	lola=lambat	sking=karena
bwt=untuk	lsg=langsung	skl=sekali
byk=banyak	ma=dengan	skr=sekarang
byr=bayar	makasih=terimakasih	skrg=sekarang
cacad=cacat	make=pakai	skrng=sekarang
cape=lelah	makenya=pakainya	slalu=selalu

capek=lelah	makin=semakin	sllu=selalu
cb=coba	maksa=paksa	slnjtny=selanjutnya
cm=cuma	males=malas	smakin=semakin
cmn=cuma	malesin=memalaskan	sm=sama
cpt=cepat	mampus=mati	smnjak=semenjak
cr=cara	maning=lagi	smp=sampai
cobain=coba	mantab=mantap	smpai=sampai
contreng=centang	mao=mau	smpe=sampai
cucok=cocok	masa=yakin	smua=semua
cuma=hanya	masak=yakin	sngt=sangat
cuman=hanya	masi=masih	spt=seperti
d=di	masukin=masuk	stelah=setelah
da=ada	masukkin=masuk	stengah=setengah
dah=sudah	matiin=mati	stlah=setelah
daftr=datar	mbok=harusnya	stlh=setelah
dapet=dapat	mcm=macam	suseh=susah
dcoba=dicoba	mdh=mudah	sy=saya
demen=suka	mentok=berhenti	tak=tidak
dech=deh	mesti=harus	tanggepan=tanggapan
dftar=daftar	mgkn=mungkin	tambh=tambah
dftr=daftar	mintip=mantap	tarok=taruh
dg=dengan	mlh=malah	tau=tahu
dgn=dengan	mmg=memang	taunya=tahunya
dgan=dengan	mnt=minta	taun=tahun
dibenerin=benar	mo=mau	td=tadi
diberesin=diselesaikan	molo=melulu	tdak=tidak
diblgny=dibilangnya	mulu=melulu	tdk=tidak
dicobain=coba	mululah=melulu	tetep=tetap
dikecewain=kecewa	ms=masih	tf=transfer
dikit=sedikit	msh=masih	tggu=tunggu
dipake=dipakai	mskin=memasukkan	thdp=terhadap
dperbaiki=diperbaiki	msknya=masuknya	thn=tahun
dperbaikin=diperbaiki	mslh=masalah	tlg=tolong
dl=dahulu	mslah=masalah	tlng=tolong
dlm=dalam	msuk=masuk	tlong=tolong
dlu=dahulu	mw=mau	tlp=telepon
dnk=dong	n=dan	tmbah=tambah
doang=saja	nambah=tambah	tnpa=tanpa
doank=saja	nan=dan	tokcer=mantap
donk=dong	nangkring=menetap	tp=tapi
dperbaiki=diperbaiki	nanya=tanya	trdekat=terdekat
dpt=dapat	nd=dan	trf=tarif
dr=dari	ndablek=bandel	trfer=transfer

drmh=dirumah	ndaftar=daftar	trima=terima
drpd=daripada	ndak=tidak	trkadang=terkadang
dtg=datang	neh=nih	trs=terus
dunk=dong	nemuin=menemukan	trsbt=tersebut
ecek=jelek	nga=tidak	trus=terus
edyan=sinting	ngabisin=habis	trnyata=ternyata
egk=tidak	ngacir=bagus	trz=terus
emang=memang	ngaco=kacau	ttp=tetap
en=dan	ngantri=antri	tuch=tuh
g=tidak	ngapain=apa	tuk=untuk
ga=tidak	ngasih=beri	ud=sudah
gada=tidak	ngatasinnya=mengatasinya	uda=sudah
gag=tidak	ngatasinya=mengatasinya	udah=sudah
gak=tidak	ngbantu=bantu	udh=sudah
gamau=tolak	ngebantu=membantu	uk=untuk
gbs=gabisa	ngebuat=buat	usah=perlu
gede=besar	ngegarap=buat	utk=untuk
gedenya=besar	ngeselin=mengesalkan	wkt=waktu
ggn=gangguan	ngikut=ikut	x=kali
gilak=gila	ngisi=isi	y=ya
gimana=bagaimana	ngrespon=respon	yah=ya
gini=begini	ngurang=berkurang	yg=yang

Lampiran 2 Stopword List

Tabel 2 Stopword List

Stopword List					
ada	cih	jadinya	mana	punya	sungguh
agar	coba	jaman	markotob	rb	supaya
ah	cukup	jaya	mas	rp	suruh
akan	cuma	jika	masih	saat	tadi
akhir	dan	juga	mbak	saja	tahu
akhirnya	dahulu	kah	melulu	sama	tahun
aku	dalam	kalau	memang	sampai	tapi
amat	dari	kali	mending	sangat	terlalu
an	deh	kalian	mengurusi	saya	terlebih
anda	dengan	kan	merasa	sayang	tersebut
apa	di	kang	mereka	sebelumnya	terutama
apk	dia	karena	minggu	sedikit	terus
atas	dimana	ke	mu	segera	terusan
atau	dll	kemarin	mulai	sehari	ternyata
atuh	dong	kembali	muncul	sejak	tetap
awalnya	dulu	kemudian	mungkin	sekali	tetapi
b	dsb	kenapa	nah	sekarang	tiap
bagaimana	eh	kesini	namun	sekedar	tiba
bagian	gan	ketika	ni	sekian	toh
bahkan	gara	kita	nih	selalu	tsb
balik	habis	kok	numpang	selama	tu
baru	haduh	ku	nya	semakin	tuh
banyak	hampir	kumaha	nyong	sementara	ty
beberapa	hanya	lagi	oh	semua	untuk
begini	hari	lah	oi	seperti	versi
begitu	harus	lain	oiya	sering	wah
berada	hellow	lama	p	serius	waktu
berapa	huh	langsung	pada	setelah	walaupun
berkali	ieu	lalu	padahal	setiap	ya
berulang	in	lebih	pakai	si	yah
bla	ini	lho	pasti	sih	yakin
bolak	ingin	loh	pernah	solusi	yang
boss	itu	lo	perlu	son	yp
bro	iya	mah	pertama	sudah	
bulan	jadi	malah	pun	sudahlah	

Lampiran 3 Daftar Kata Dasar

Tabel 3 Daftar Kata Dasar

Kata Dasar					
ada	bukan	jawab	luncur	perlu	sesuai
admin	bulan	jelas	maki	pilih	sewa
akses	bulu	jengkel	makin	potong	siksa
aktivasi	butuh	jos	maksa	pribadi	sila
alami	cekat	jumlah	maksimal	puas	singkat
aman	cepat	kadang	maksud	puluh	sinyal
ampun	coba	kapan	malas	putus	soal
andai	daftar	karena	malu	rasa	solusi
antar	dapat	kasih	mandiri	rekan	sulit
apa	dasar	kata	mantab	repot	suruh
aplikasi	dekat	kecewa	manfaat	respon	susah
arti	depan	kecil	masalah	rugi	tahun
atasi	digit	kelas	masuk	salah	tambah
atur	dikit	keluar	menang	sambung	tampil
baca	dukung	keluh	merah	saran	tanggap
bagai	emosi	kembang	mesin	sebab	tarik
bagus	fitur	kena	meski	sebal	terima
baik	fungsi	kenapa	milik	sedang	terimakasih
bank	ganggu	kencang	minggu	sederhana	teknis
bantu	ganti	kendali	mudah	sedih	teknologi
banyak	genggam	kepala	murah	sedikit	telah
baru	guna	keren	mutasi	sedot	tempat
bayar	habis	kerja	nasib	segala	temu
becus	hadap	kesal	nilai	sekali	tentu
begini	harap	ketik	nyaman	sekarang	tetap
begitu	harus	ketika	nyata	selesai	tinggal
beli	hasil	kirim	operasi	selalu	tingkat
belum	henti	koneksi	pakai	selingkuh	tinggi
benah	hilang	kurang	paksa	semua	tolong
benar	hubung	lain	panjang	sempurna	transaksi
berantak	iklan	laksana	pantau	sendat	tugas
berapa	ikut	lama	parah	sendiri	tujuan
beres	informasi	lambat	partisipasi	sentuh	tulis
besar	instalasi	lancar	pasti	seperti	tumpuk
betul	izin	langgan	payah	serah	tutup
biasa	jaga	langkah	pecah	serba	ubah
bijak	jalan	lanjut	pecat	serius	urus
bilang	jamin	layan	penting	serta	usaha
buat	jangan	lelah	penuh	service	versi

Lampiran 4 Daftar Kata Positif

Tabel 4 Daftar Kata Positif

Term Posistif				
aman	cerdas	hebat	mutu	sip
akurat	dapat	informatif	normal	stabil
bagus	dukung	jelas	nyaman	suka
baik	efektif	jos	puas	sukses
bantu	efisien	keren	penting	tanggap
beres	enak	lancar	praktis	tegas
becus	fungsi	lengkap	respon	top
bisa	gahar	manfaat	responsif	
cantik	gampang	mantap	sempurna	
cekat	guna	mudah	simpatik	
cepat	hasil	mulus	senang	

Lampiran 5 Daftar Kata Negatif

Tabel 5 Daftar Kata Negatif

Term Negatif				
bandel	gabisa	kecewa	mohon	sampah
becus	gamau	keluh	murah	sebal
bingung	ganggu	kendala	nyangkut	sedih
buruk	gembel	kesal	parah	sendat
busuk	gila	lama	pecah	sesal
brengsek	hancur	lambat	rendah	siksa
cacat	hambat	lelah	repot	sinting
cekal	jelek	malas	rugi	suram
cuek	jengkel	malu	runyam	susah
gagal	kacau	masalah	rusak	tolol

Lampiran 6 Pelabelan Data Bagian 1

Nama: Alief Maulana
NIM: 115120401111008

Tgl: 28 April 2015
TTD: Alief

No	Review	Positif	Negatif
411	SLoved it Loved it.. Membantu sekali	✓	
412	Tidak Bisa Entry m-bca PIN waktu awal akses, diminta membuat pin untuk akses m-bca, 6 digit dgn minimal satu alphabet. setelah itu, muncul window entry pin, tapi kok yg muncul keypad angka saja ya? ga bsa entry huruf.. jadi ga bsa masuk m-bca nya... please fix..		✓
413	Berfungsi dengan baik Sudah bagus ini sekarang. Tampilan klo bisa android banget donk.	✓	
414	Aplikasinya "GJ" alias Ga Jelas..error melulu...		✓
415	It works..!! Berhasil install dan transaksi.. thanks..	✓	
416	Jelek Kenapa selalu error 101 - cannot identify your login autentification. Masak gagal teru. Pdhl aplikasi ini berguna banget, tp lbh byk gagal loginny. Coba dperbaiki lagi. Kenapa sekarang malah gak bisa login, disuruh daftar lewat atm. Payah nih, aplikasi cacat.		✓
417	aplikasi rusak!!! hapus aja! gak ada gunanya! huh!		✓
418	Error Error..... mululah..107 dsb...		✓
419	Fucklah bca Udah 4 bln install unistall terus2an ga bs parsing, masukin kode atm aja sending mulu, kacau deh, malu2in aja iih, knp d android pd bermasalah, kalo d bb lancar aja, gimana ni bca gadaa tanggepan sm skl, kayak ga mutu app nya, sucks		✓
420	Payah.. Apa apaan nih. Cuman launcher doang.. Anak SMK juga bisa bikin beginian.. uninstall..		✓

Lampiran 7 Pelabelan Data Bagian 2

Nama: Ita Anggraini
NIM: 115080300111025

Tgl: 1 Mei 2015
TTD:

No	Review	Positif	Negatif
411	SLoved it Loved it.. Membantu sekali		✓
412	Tidak Bisa Entry m-bca PIN waktu awal akses, diminta membuat pin untuk akses m-bca, 6 digit dgn minimal satu alphabet. setelah itu, muncul window entry pin, tapi kok yg muncul keypad angka saja ya? ga bsa entry huruf.. jadi ga bsa masuk m-bca nya... please fix..	✓	
413	Berfungsi dengan baik Sudah bagus ini sekarang. Tampilan klo bisa android banget donk.	✓	
414	Aplikasinya "GJ" alias Ga Jelas..error melulu...		✓
415	It works..!! Berhasil install dan transaksi.. thanks..	✓	
416	Jelek Kenapa selalu error 101 - cannot identify your login autentification. Masak gagal teru. Pdhl aplikasi ini berguna banget, tp lbh byk gagal loginny. Coba dperbaiki lagi. Kenapa sekarang malah gak bisa login, disuruh daftar lewat atm. Payah nih, aplikasi cacat.	✓	
417	aplikasi rusak!!! hapus aja! gak ada gunanya! huh!		✓
418	Error Error..... mululah..107 dsb...		✓
419	Fucklah bca Udah 4 bln install unistall terus2an ga bs parsing, masukin kode atm aja sending mulu, kacau deh, malu2in aja iih, knp d android pd bermasalah, kalo d bb lancar aja, gimana ni bca gadaa tanggepan sm skl, kayak ga mutu app nya, sucks		✓
420	Payah.. Apa apaan nih. Cuman launcher doang.. Anak SMK juga bisa bikin beginian.. uninstall..		✓

Lampiran 8 Pelabelan Data Bagian 3

Nama: Pasha T. W. **Tgl:** 28-09-2015
NIM: 145150209111016 **TTD:**

No	Review	Positif	Negatif
411	SLoved it Loved it.. Membantu sekali	✓	
412	Tidak Bisa Entry m-bca PIN waktu awal akses, diminta membuat pin untuk akses m-bca, 6 digit dgn minimal satu alphabet. setelah itu, muncul window entry pin, tapi kok yg muncul keypad angka saja ya? ga bsa entry huruf.. jadi ga bsa masuk m-bca nya... please fix..		✓
413	Berfungsi dengan baik Sudah bagus ini sekarang. Tampilan klo bisa android banget donk.	✓	
414	Aplikasinya "GJ" alias Ga Jelas..error melulu...		✓
415	It works..!! Berhasil install dan transaksi.. thanks..	✓	
416	Jelek Kenapa selalu error 101 - cannot identify your login autentification. Masak gagal teru. Pdhl aplikasi ini berguna banget, tp lbh byk gagal loginny. Coba dperbaiki lagi. Kenapa sekarang malah gak bisa login, disuruh daftar lewat atm. Payah nih, aplikasi cacat.		✓
417	aplikasi rusak!!! hapus aja! gak ada gunanya! huh!		✓
418	Error Error..... mululah..107 dsb...		✓
419	Fucklah bca Udah 4 bln install unistall terus2an ga bs parsing, masukin kode atm aja sending mulu, kacau deh, malu2in aja iih, knp d android pd bermasalah, kalo d bb lancar aja, gimana ni bca gadaa tanggepan sm skl, kayak ga mutu app nya, sucks		✓
420	Payah.. Apa apaan nih. Cuman launcher doang.. Anak SMK juga bisa bikin beginian.. uninstall..		✓