KLASIFIKASI *REVIEW* PRODUK KECANTIKAN PADA APLIKASI SOCIOLLA MENGGUNAKAN ALGORITME *MODIFIED K-NEAREST NEIGHBOR* (MK-NN) DENGAN PEMBOBOTAN BM25

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Alfita Nuriza
NIM: 165150201111095

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2020

PENGESAHAN

KLASIFIKASI *REVIEW* PRODUK KECANTIKAN PADA APLIKASI SOCIOLLA MENGGUNAKAN ALGORITME *MODIFIED K-NEAREST NEIGHBOR* (MK-NN) DENGAN PEMBOBOTAN BM25

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer

Disusun Oleh:

Alfita Nuriza

NIM: 165150201111095

Skripsi ini telah diuji dan dinyatakan lulus pada

23 Juli 2020

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I	Dosen Pembimbing II
Indriati, S.T., M.Kom	Nurul Hidayat, S.Pd., M.Sc
NIP: 19831013 201504 2 002	NIP: 19680430 200212 1 001

Mengetahui

Ketua Jurusan Teknik Informatika

Achmad Basuki, S.T., M.MG., Ph.D.

NIP: 19741118 200312 1 002

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk mendapat gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi naskah ini dan disebutkan dalam daftar referensi.

Apabila ternyata dalam 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 70).

Malang, 25 Juni 2020

Alfita Nuriza

NIM: 165150201111095

ABSTRAK

Alfita Nuriza, Klasifikasi *Review* Produk Kecantikan Pada Aplikasi Sociolla Menggunakan Algoritme *Modified K-Nearest Neighbor* (MK-NN) dengan Pembobotan BM25

Pembimbing: Indriati, S.T., M.Kom. dan Nurul Hidayat S.Pd., M.Sc.

Produk kecantikan telah menjadi salah satu dari sekian banyak hal yang tidak dapat lepas dari kaum wanita karena tuntutan untuk tampil cantik serta menarik. Berbagai produk tersebut menawarkan keunggulan – keunggulan nya, ada banyak produk kecantikan di pasaran, mulai dari perawatan kulit dan kosmetik dari berbagai jenis dan merek. Produk-produk ini memiliki kelebihan, tetapi tidak semua produk memenuhi kebutuhan penggunanya. Hal ini adalah sesuatu yang harus diperhatikan konsumen sebelum membeli. Di sisi lain dengan Jumlah produk kecantikan yang banyak terkait erat dengan pendapat tentang produk tertentu sesuai dengan parameter yang diberikan oleh konsumen seperti kelebihan, kekurangan, kualitas dan parameter lainnya., hal inilah yang digunakan sebagai referensi. Salah satu *platform* perdagangan elektronik yang menyediakan produk kecantikan adalah *Sociolla.* Buakan hanya menjual produk kecantikan, pada *platform* ini juga terdapat ulasan atau *review* dari konsumen. Dengan membaca semua *review* tersebut secara lengkap akan menyita banyak waktu, sedangkan jika hanya membaca sedikit, evaluasi yang dihasilkan akan menjadi bias. Untuk mengatasi permasalahan tersebut dilakukan klasifikasi dari *review* yang ada yang akan diklasifikasikan ke dalam 2 kelas yaitu kelas positif serta negatif. Dalam penelitian ini penulis menggunakan algoritme *Modified K-Nearest Neighbor* (MK-NN) dengan BM25 sebagai pembobotan. Data yang dipakai sejumlah 500 data yang terbagi menjadi dua yaitu positif dan negatif. Dari hasil evaluasi pengujian dengan 5-*fold cross validation* dihasilkan rata-rata nilai akurasi, *precision, recall, dan f-measure* tertinggi sebesar 51,00%, 50,90%, 52,61%, dan 51,70% pada saat nilai *k*=11.

Kata kunci: *klasifikasi, Produk Kecantikan,Text Mining, Modified K-Nearest Neighbor, BM25.*

ABSTRACT

Alfita Nuriza, Beauty Product Review Classification in Sociolla Application Using K-Nearest Neighbor (MK-NN) Algorithm with BM25 Weighting.

Supervisors: Indriati, S.T., M.Kom. and Nurul Hidayat S.Pd., M.Sc.

Beauty products have become one of the many things that cannot be separated from women because of the demands to look beautiful and attractive. These products offer their advantages, there are many beauty products on the market, ranging from skin care and cosmetics from various types and brands. These products have advantages, but not all products meet the needs of its users. This is something that consumers must pay attention to before buying. On the other hand, the number of beauty products that are closely related to opinions about certain products in accordance with the parameters given by consumers such as strengths, weaknesses, quality and other parameters, this is what is used as a reference. One electronic trading platform that provides beauty products is Sociolla. Not only sell beauty products, on this platform there are also reviews from consumers. Reading all these reviews in full will take up a lot of time, whereas if you only read a little, the resulting evaluation will be biased. To overcome these problems the classification of the existing review will be classified into 2 classes, namely positive and negative classes. In this study the authors used the Modified K-Nearest Neighbor (MK-NN) algorithm with BM25 as a weighting. The data used were 500 data which were divided into two, positive and negative. From the evaluation results of the test with 5-fold cross validation, the highest average values of accuracy, precision, recall, and f-measure were 51.00%, 50.90%, 52.61%, and 51.70% at the time k = 11.

Keywords: *Classification, Beauty Product, Text Mining, Modified K-Nearest Neighbor, BM25*

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR PERSAMAAN

BAB 1 PENDAHULUAN

Pada bab pendahuluan akan dijabarkan tentang apa yang dikerjakan dalam penelitian ini yang mencakup latar belakang yang berisi bagaimana munculnya ide sehingga dapat dianalisis permasalahan yang ada dalam penelitian ini, kemudian terdapat rumusan masalah yang berisi pertanyaan yang mendorong penulis untuk menjawabnya, tujuan yang berisi tujuan yang ingin dicapai pada skripsi ini, manfaat berisi dampak dari penelitian ini dalam lingkup yang lebih luas, batasan masalah untuk membantu menjabarkan ruang lingkup dari permasalahan penelitian, dan yang terakhir sistematika pembahasan yaitu berisi struktur dari penelitian ini.

1.1 Latar Belakang

Dalam kehidupan sehari-hari kaum perempuan tidak lepas dari tuntutan untuk tampil cantik dan juga menarik. Produk kecantikan telah menjadi salah satu hal yang tidak lepas dari kaum wanita. Tidak heran jika di Indonesia perkembangan industri kecantikan terus berkembang tiap tahunnya. Dilansir dari data yang diperoleh dari Kemenperin atau Kementrian Perindustrian pada tahun 2017, terdapat lebih dari 760 perusahaan di bidang industri *make-up* Indonesia. Kemenperin menargetkan bahwa pada tahun ini industri kosmetik dan kecantikan naik sebesar 9% dibanding tahun sebelumnya yaitu 7,3%. Hal ini menunjukkan bahwa tren dari kebutuhan masyarakat terhadap produk kecantikan sangatlah tinggi. Menurut survei yang dilakukan oleh ZAP Beauty Index pada tahun 2018 yang melibatkan sebanyak 17.889 responden wanita di Indonesia, menunjukkan bahwa sejak usia kurang dari 18 tahun (13-15 tahun) dengan presentase 41,9% sudah mengenal *make up*. Sementara itu menurut data yang sama, sebesar 36,4% remaja yang berusia 13-15 tahun sudah melakukan perawatan di klinik kecantikan (Syifa, 2018).

Seiring dengan tingginya permintaan calon pembeli terhadap produk kecantikan dan dengan didukung oleh perkembangan zaman yang sudah canggih, memunculkan beberapa *platform* atau *e-commerce* yang menjual berbagai produk kecantikan baik itu *make up* maupun *skin care*. Salah satunya adalah Sociolla yang sudah berdiri pada tahun 2015. Tidak hanya terfokus pada produk kecantikan saja, namun Sociolla juga menawarkan fitur-fitur lain berupa media informasi yang terdapat cara pemakaian produk yang benar, serta beberapa jurnal yang di buat oleh tim Sociolla untuk memberikan kesan terlibat langsung kepada pengguna *platform* mereka. Dengan portal web *review.soco.id* yang memuat berbagai artikel kecantikan, *review* produk dari para konsumen.

Sebelum membeli produk-produk kecantikan ataupun alat kecantikan di *sociolla*, konsumen sebaiknya mencari tahu akan detail produk yang mereka beli, hal ini dapat di telusuri melalui review atau opini yang di kemukakan oleh para penggguna *Sociolla* di kolom ulasan produk serta *Beauty Journal.* Dengan adanya *Beauty Journal* yang dibuat oleh *Sociolla* tersebut dapat digunakan sebagai referensi dan acuan untuk menggunakan suatu produk tertentu apakah layak atau

tidak layak digunakan. Konsumen tidak hanya dapat memesan produk melalui aplikasi tetapi juga dapat memberi ulasan produk pada aplikasi tersebut. Namun dengan membaca semua ulasan atau review secara tidak langsung dinilai kurang efisien karna memakan banyak waktu, namun jika hanya membaca sedikit beberapa ulasan maka bisa akan didapat hasil yang bias. Klasifikasi bertujuan untuk mengatasi permasalahan yang telah dijelaskan, dengan mengelompokkan opini pengguna menjadi opini positif atau negatif. Selain itu pihak perusahaan juga membutuhkan *feedback* terhadap produk mereka, semakin berkembang produknya biasanya semakin banyak pula opini-opini dari masyarakat mengenai produk mereka. Disini kualitas dari suatu produk dapat di tentukan dari *review* yang baik dari pembeli yang telah merasakan produknya sehingga produsen juga dapat mengkategorikan *review* atau ulasan yang ada sehingga diharapkan produk yang diciptakan dapat dievaluasi dan diperbaiki kekurangannya.

Untuk menghasilkan opini negatif dan positif dari hasil klasifikasi di butuhkan suatu metode *classifier*. Terdapat beberapa metode yang biasa digunakan untuk pengklasifikasian, salah satunya adalah metode *K-Nearest Neighbor* yang merupakan salah satu dari 10 metode yang paling terkenal, sedangkan metode *Modified KNN* yang merupakan metode klasifikasi hasil pengembangan dari metode KNN karena metode KNN biasa dinilai memiliki bebrapa kekurangan. Metode *Modified K-Nearest Neighbor* memiliki akurasi sebesar 99,51%, jika di bandingkan dengan metode *K-Nearest Neighbor* saja terdapat perbedaan akurasi sebesar 5-7% (Gazalba, et al., 2017).

Metode BM25 merupakan metode yang digunakan untuk mencari nilai kesamaan antar dokumen. Menurut (Tinega, et al., 2018) BM25 dinilai jauh lebih baik jika dibandingkan dengan *Vector Space Model* dan *Boolean Model.*

Berdasarkan uraian yang telah dijelaskan dapat dilakukan penelitian menggunakan metode MK-NN dan BM25 yang mengacu pada penelitian-penelitian terdahulu sehingga diharapkan mampu meghasilkan nilai akurasi yang cukup tinggi. Diharapkan dengan adanya hasil dari klasifikasi ini, dapat memberikan dampak yang baik terhadap konsumen berupa kemudahan dalam mencari dan memilih referensi produk kecantikan yang akan dibeli dan menjadikan opini dari konsumen sebagai bahan evaluasi bagi produsen sehingga menghasilkan produk yang lebih baik. Dengan beberapa referensi pendukung yang diinginkan penulis dalam menulis karya ilmiah maka penulis mengusulkan penelitian yang berjudul Klasifikasi *Review* Produk Kecantikan Pada Aplikasi *Sociolla* Menggunakan Algoritma *Modified K-Nearest Neighbor* (MK-NN) dan BM25.

1.2 Rumusan Masalah

Mengacu pada latar belakang penelitian ini, dapat di jelaskan masalah yang ada di penelitian ini sebagai berikut:

1. Bagaimana pengaruh nilai *k* terhadap hasil evaluasi pengujian algoritme *Modified K-Nearest Neighbor* dan BM25?

2. Bagaimana tingkat akurasi yang dihasilkan menggunakan algoritme *Modified K-Nearest Neighbor* dan BM25 pada review produk kecantikan di aplikasi Sociolla

1.3 Tujuan

Dengan penjabaran rumusan masalah, maka tujuan penelitian ini adalah:

1. Mengimplementasikan algoritme *Modified K-Nearest Neighbor* dan BM25 untuk mengklasifikasikan *review* produk kecantikan di aplikasi Sociolla.
2. Menguji tingkat akurasi yang didapatkan dari metode *Modified K-Nearest Neighbor* dan BM25 dalam melakukan klasifikasi pada *review* produk kecantikan di aplikasi Sociolla.

1.4 Manfaat

Penelitian ini dapat memberikan manfaat antara lain:

1. Konsumen khususnya konsumen produk kecantikan dapat terbantu dalam memilih atau mencari produk apa yg cocok digunakan dan yang tidak dengan melihat sentimen analisis yang dihasilkan.
2. Membantu pihak produsen untuk mengetahui respon konsumen sehingga diharapkan dapat meningkatkan kualitas produk nya dan mengetahui kekurangan dan mengevaluasinya.

1.5 Batasan Masalah

Berikut merupakan batasan masalah penelitian yang ditujukan untuk menjelaskan cakupan ruang lingkup masalah pada penelitian ini sehingga mudah untuk di mengerti:

1. Sumber data yang digunakan berasal dari aplikasi *Sociolla*, data tersebut berupa *review* pada produk Nature Republic Aloe Vera Soothing Gel. Produk ini dipilih karena merupakan salah satu produk yang memiliki *review* paling banyak.
2. Data yang dipakai sejumlah 500 data yang terdiri dari 250 data posistif dan 250 data negatif.
3. Data yang dipakai menggunakan Bahasa Indonesia.

1.6 Sistematika Pembahasan

Bagian yang merupakan format dari keseluruhan penelitian ini yang berisikan 7 bab, pada sub bab ini terdapat deskripsi singkat dari beberapa bab yang membantu pembaca penelitian ini untuk memahami intisari serta sistematika di dalam peneitian ini.

BAB 1 Pendahuluan

Pada bab pendahuluan akan dijabarkan dasar dilakukannya penelitian ini atau latar belakang terjadinya permasalahan yang muncul, rumusan masalah yang ingin diselesaikan, penelitian bertujuan untuk apa, manfaat penelitian yang di peroleh, batasan masalah penelitian yang mencakup ruang lingkup penelitian ini, serta gambaran umum dan sistematika penelitian agar memberikan gambaran secara umum untuk penelitian yang telah dilakukan.

BAB 2 Landasan Kepustakaan

Pad bagian ini, terdapat referensi-referensi yang menjelaskan tentang penelitian yang sudah di lakukan sebelumnya terkait dengan klasifikasi. Kemudian terdapat penjelasan mengenai dasar teori yang secara langsung berkaitan dengan penelitian yang di lakukan

BAB 3 Metodologi Penelitian

Pada bab metodologi ini menunjukkan secara terperinci mengenai urutan prpses yang akan dilakukan pada penelitian ini.

BAB 4 Perancangan

Bab ini berisi tentang bentuk perancangan sistem yang akan diterapkan sehingga dapat mengklasifikasikan suatu opini berupa *review* produk serta sistem penelitian yang telah di buat dengan rancangan sistem penelitian secara keseluruhan.

BAB 5 Implementasi

Bab ini memberikan kejelasan yang akurat dan mendalam mengenai pengimplementasian algoritma yang ada pada penelitian ini dan pembahasan klasifikasi review produk atau ulasan produk kecantikan yang ada di aplikasi *Sociolla.*

BAB 6 Pengujian dan Analisis

BAB 6 berisi tentang hasil dari pengujian yang sudah dirancang sebelumnya dan juga dibahas mengenai analisis terhadap hasil yang diperoleh dalam pengujian ini.

BAB 7 Penutup

Bab ini menjelaskan penelitian yang disimpulkan penulis dan beberapa saran yang bisa digunakan sebagai pertimbangan guna penelitian selanjutnya yang bisa dikembangkan lebih baik lagi.

BAB 2 LANDASAN KEPUSTAKAAN

Pada bagian ini mengandung beberapa penelitian yang telah di lakukan oleh peneliti terdahulu dengan teori serta metode yang di gunakan agar dapat di analisa sebagai rujukan pustaka.

2.1 Kajian Pustaka

Hasil dari riset terdahulu yang ada kaitannya dengan penelitian ini terkait dengan klasifikasi, metode MK-NN dan juga metode BM25. Penelitian tersebut merupakan karya ilmiah yang berupa skripsi, jurnal serta tesis untuk pemberian kajian pustaka dan referensi studi literatur yang menjadi dasar untuk memecahkan masalah dan penanggulangannya.

Metode MK-NN dipilih penulis karena berdasarkan kajian pustaka yang telah dibaca di bandingkan dengan KNN menghasilkan nilai dan akurasi yang lebih baik. Dikutip dari jurnal "Comparative Analysis of K-Nearest Neighbor and Modified K-Nearest Neighbor Algorithm for Data Classification" (Okfalisa, Ikbal Gazalba, Mustakim, Nurul Gayatri Indah Reza, 2017) penelitian tersebut membandingkan antara beberapa metode yaitu *metode KNN* dengan metode *Modified KKN* yang mana hasilnya metode *Modified KKN* lebih baik dnegan hasil akurasi 99,20% di bandingkan metode *KKN* yang hanya 93,94% akurasinya.

Beberapa riset yang membahas tentang topik analisasi sentimen telah marak dilakukan pada penelitian terdaulu, contohnya yang telah di lakukan oleh Indriya Dewi Onantya, Indriati, dan Putra Pandu Adikara yang berjudul "Analisis Sentimen Pada Ulasan Aplikasi BCA Mobile Menggunakan BM25 Dan Improved K-Nearest Neighbor". Pada riset yang dilakukan ini data yang diguanakan sebanyak 400 data *training* dan 100 data *Testing* yang mana hasil akhirnya menghasilkan 2 klasifikasi kelas yang mana terdiri dari kelas negatif dan kelas positif. Dari penelitian ini dihasilkan nilai pengujian evaluasi menggunakan pengujian *5-fold* dengan nilai k tertinggi saat nilai k=10 dengan hasil nilai *precision* 0.946, *recall* 0,934, serta *accuracy* 0,942.

Selanjutnya juga contoh penelitian yang terkait dilakukan oleh Ardhimas Ilham Bagus Pranata pada tahun 2019 yang menggunakan metode BM25 untuk menghitung kemiripan dokumen dan di kombinasikan dengan Algoritme *Improved K-Nearest Neighbor* untuk memberikan klasifikasi dokumen pada laporan kepolisisan. Data yang di gunakan sejumlah 100 data dan dihasilkan nilai akurasi sebesar 95%. Literatur yang digunakan terdapat pada Tabel 2.1 .

Tabel 2.1 Daftar Kajian Pustaka

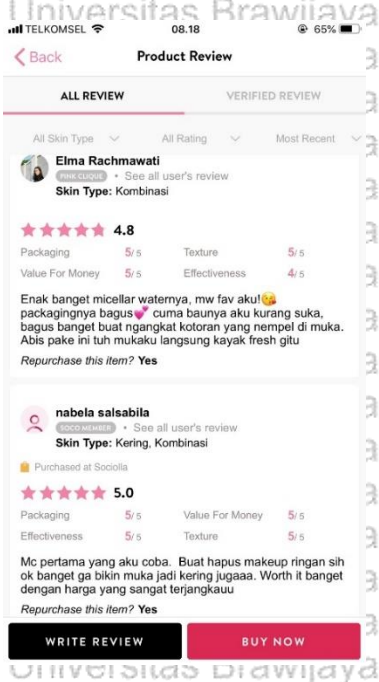
No.	Pustaka	Objek	Metode	Hasil
1	(Okfalisa, Gazalba & Mustakim, 2017)	Data Unit Pelaksana Transfer Tunai Bersyarat (Unit Pelaksana Program Keluarga Harapan)	-*K-Nearest Neighbor* -*Modified K-Nearest Neighbor*	Uji hasil dengan *K-Fold Validation* menghasilkan hasil evaluasi berupa nilai rasio pada saat metode MK-NN sebesar 99,51% sedangkan penggunaan metode K-NN hanya sebesar 94,95%
2	(Onantya, Indriati & Adikara, 2019)	ulasan pada aplikasi BCA *mobile*	-BM25 -*Improve K-Nearest Neighbor*	Nilai uji hasil evaluasi uji 5-*fold cross validation* mendapat hasil dari nilai k terbaik saat nilai k sebesar 10 dengan nilai *precision* 0,946, recall 0,934, *f-measure* 0,939, dan *accuracy* 0,942
3.	(Royyan,2018)	Ulasan pada aplikasi telepon genggam (*mobile banking*)	*Modified KNN*	Akurasi dengan nilai tertinggi yang diperoleh adalah 76% pada saat k dengan nilai 11 serta data total latih= 400, namun untuk data latih yang sejumlah 200 data, tingkat dari nilai akurasi tertingginya berbeda yaitu 69% (K=3). Dan 70% untuk 300 data latih dan K=3.
4.	(Pranata, Indriati & Marji,2019)	Lampiran serta data dokumen laporan kepolisian	-BM25 -*Improved K-Nearest Neighbor* (IKNN)	Nilai hasil pada evaluasi pengujian menggunakan *K-fold* memberikan nilai tertinggi jika nilai k=15 berupa *precision*=0,953373, *recall*=0,931382, f-*measure*=0,938122 dan *accuracy*=0,956795

5.	(Binawan, Indriati & Muflikhah, 2019)	Karya ilmiah pada bagian abstrak dokumen	*Modified K-Nearest Neighbor*	Pencantuman nilai dengan hasil terbaik didapatkan pada saat nilai *k*=11 dengan nilai *f-measure* sebesar 0,9092, *recall* sebesar 0,9087, dan *precision* sebesar 0,9265.

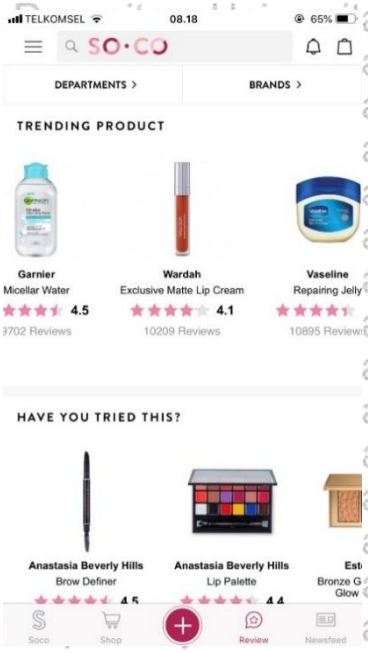
2.2 Produk Kecantikan

Secara harfiah produk adalah hasil dari usaha seorang individu atau kelompok dalam menciptakan barang ataupun jasa yang mana di samping itu juga melewati berbagai macam rangkaian proses untuk menghasilkan produk jadi. Dengan kata lain suatu produk yang berupa barang ataupun jasa dapat menguntungkan melalui suatu transaksi atau transaksi perdagangan. Pengertian produk juga dapat di kaitkan dengan segala sesuatu yang ada ataupun tersedia, dapat dirasakan, dimiliki, konsumsi, serta di gunakan untuk dapat memenuhi kebutuhan dari pada fisik manusia, tempat, jasa, manusia, beberapa gagasan, dan organisasi. (Kotler & Amstrong, 2001).

Menurut sebuah penelitian yang dilakukan oleh lembaga penelitian pada awal Mei 2017, dalam sebuah penelitian yang melibatkan 1.200 responden, kecantikan sudah dikenal luas, dan kebanyakan orang mendefinisikan kecantikan sebagai penampilan fisik. Dalam penelitiannya, Sigma Research membagi definisi kecantikan menjadi tiga jenis penilaian, yaitu kecantikan, otak, dan perilaku. Kecantikan adalah penilaian fisik, otak didasarkan pada penilaian kecerdasan, dan perilaku adalah mendefinisikan kecantikan melalui penilaian perilaku. Melalui wawancara dengan 1.200 orang, lebih dari 40% mendefinisikan kecantikan sesuai dengan kondisi fisik mereka. Hanya 14,8% orang yang mendefinisikan kecantikan dengan kepribadian yang menarik, sementara 9,5% berpikir bahwa perilaku ramah itu indah. Meskipun kecerdasan tampaknya tidak dianggap sebagai salah satu kualitas kecantikan yang menentukan, Karena hanya 6,1% orang yang berpikir bahwa orang pintar adalah orang yang cantik. Oleh karena itu dapat disimpulkan bahwa produk kecantikan dimaksudkan untuk digunakan baik pada membersihkan kulit serta mengencangkan, meningkatkan daya tarik, meningkatkan kecantikan, serta ada umumnya untuk mengubah penampilan namun produk kecantikan tidak lebih dari ini, sehingga tidak mengubah struktur atau fungsi dari tubuh seorang manusia. Produk kecantikan yang ada di aplikasi *Sociolla* ini bisa di lihat di Gambar 2.1 diikuti dengan *review* di tiap produknya pada Gambar 2.2



Gambar 2.1 *Review* dari produk



Gambar 2.2 Produk yang ada di aplikasi Sociolla

2.3 Klasifikasi

Dalam hal ini, yang di maksudkan dengan Klasifikasi adalah proses beberapa dokumen ke dalam kelas tertentu. Klasifikasi sangat berguna untuk menganalisis opini dari masyarakat, termasuk opini berupa *review* produk kecantikan. Selain menganalisis hasil opini tersebut, hal yang biasanya ingin orang ketahui adalah apa yang menyebabkan sentimen tersebut dan kapan hasil dari sentimen akan berubah, karena dengan mengetahui hasilnya, pihak yang bersangkutan dapat memperbaiki hal hal yg membuat hasil dari opini masyarakat menurun. Hal ini membuat klasifikasi sangat penting karena akan berguna bagi perusahaan, organisasi maupun individu dalam meraih kesuksesan. Besarnya manfaat dan pengaruh dari klasifikasi menyebabkan penelitian maupun aplikasi mengenai klasifikasi berkembang pesat.

2.4 Text Mining

Text mining adalah metode klasifikasi yang merupakan varian dari *data mining* yang tujuannya untuk menemukan suatu pola yang menarik dari data tekstual yang berjumlah besar (Han, et al., 2006). Namun tidak hanya itu pada klasifikasi, *text mining* juga berguna dalam menyelesaikan masalah berupa *information extraction*, *clustering,* dan *information retrival*. Penggunaan *Text mining* juga dapat memudahkan beberapa hal efisien dalam mengidentifikasi sebuah tren di berbagai bidang, serta pengunaan dari *text mining* juga dapat mendeteksi proses pegiat plagiasi atau duplikasi. *Text mining* juga di gunakan untuk mengambil secara otomatis atau mencari kekurangan antara kode program serta dokumentasinya secara otomatis.

2.4.1 Pre-Processing

Tahap ini merupakan awal dari data yang akan masuk pada proses klasifikasi, yang mana terdapat penekanan khusus untuk menghilangkan serta mengatasi *noisy data,* termasuk di dalamnya berupa penyelesaian data informasi yang hilang serta tidak lengkap (Adiwijaya, 2006). Tahap awal Pre-processing ini juga di tujukan untuk merepresentasikan pada beberapa dokumen sebagai *vector fitur*, yaitu teks yang tertera ada pencarian dapat dikatakan di pisah menjadi kata individual untuk hasil dan tujuannya. Keywords dalam pencarian ini merupakan salah satu pemilihan fitur, yang dapat di katakan sebagai langkah pendekatan untuk pembuatan daftar kata dokumen (Srividhya dan Anitha, 2010). Tahapan pada proses awal *pre-processing* ini terdirir dari *case folding, tokenizing, filtering, dan stemming.*

2.4.2 Case Folding

Tahap *Case folding* pada pencarian haruf menggunakan huruf kecil dari penggunaan huruf capital sebelumnya (Sanjaya, 2017).

2.4.3 Tokenisasi

Tokenisasi dapat digunakan untuk memisahkan kalimat dalam dokumen kata demi kata untuk memudahkan langkah selanjutnya yang terdiri dari *filtering* dan *stemming* (Sanjaya, 2017)

2.4.4 Filtering

Tahapan ini menggunakan *stoplist*, atau dengan kata lain merupakan meode yang sejatinya berisi kata-kata yang sering bermunculan namun tidak ada makna yang berkaitan, dengan stopwords dapat menghilangkan kata-kata yang tidak ada maknanya tersebut tidak terkait dengan karakteristik untuk membedakan tiap dokumen (Sanjaya, 2017). *Stoplist yang* digunakan merupakan *stoplist* dari Tala.

2.4.5 Stemming

Stemming merupakan langkah untuk merubah *term-term* ke kata dasar, dengan cara membuang imbuhan, awalan dan akhiran(Sanjaya, 2017).

2.5 K-Nearest Neighbor

Tahap K-Nearest Neighbor, merupakan salah satu meode baik di gunakan untuk klasifikasi dengan cara mencari tetangga terdekat dari data yang ingin diklasifikasikan dengan cara mencari kelas terbanyak dari kumpulan sejumlah k dari dokumen yang paling mirip dengan data latih. Kelas yang paling banyaklah yang digunakan sebagai kelas dari data uji (Rachmat & Delima, 2014). Metode ini dalam melakukan klasifikasi menggunakan cara menghitung jarak terdekat pada objek baru dengan menggunakan rumus *Euclidean* (Gazalba, et al., 2017).

2.6 Modified K-Nearest Neighbor

Metode ini merupakan metode yang menghasilkan akurasi yang baik, metode *Modified K-Nearest Neighbor* (MKNN) merupakan pengembangan dari metode konvensional KNN yang mana di anggap perlu adanya perbaikan. Perbedaan pada MKNN dengan KNN biasa yaitu dilakukannya penambahan proses validitas beberapa data latih dengan proses *weight voting* sehingga hasil kinerja metode semakin baik dengan akurasi hasil ketetanggan antar data training yang kuat. Akurasi yang lebih baik dari modifikasi yang ada pada metode MKNN di banding metode KNN biasa merupakan hasil analisa yang baik, berikut tahapannya dengan klasifikasi algoritme MK-NN:

1. Menetukan nilai k tetangga terdekat.
2. Menghitung jarak *Euclidean* antar data latih menggunakan persamaan 2.1

$$d(i,j) = \sqrt{\Sigma_{i=1}^{n}(x_{in} - x_{jn})^2} \quad (2.1)$$

Dimana:

$\mathcal{X}_{in}$ = Fitur ke-n pada data ke-i dari data uji/testing

$\mathcal{X}_{jn}$ = Fitur ke-n pada data ke-j dari data latih/training

$d(i,j)$ = Jarak dari data ke-i hingga ke-j

3. Menghitung nilai validitas data latih.

Nilai validitas data latih tergantung pada tetangga terdekatnya, yang digunakan untuk menghitung jumlah titik dengan label yang sama untuk data tersebut. Untuk menghitung nilai validitas data dapat dilihat pada persamaan 2.2

$$Validity(x) = \frac{1}{H}\Sigma_{i=0}^{n} S(lbl(x), lbl(Ni(x))) \quad (2.2)$$

Dimana:

H : jumlah titik terdekat

$lbl(x)$: kelas x

$lbl(Ni(x))$: label kelas titik terdekat x

Dengan ini Fungsi dari S dipakai untuk menghitung titik x dengan data ke-i yang sama dari tetangga paling dekat. Yang ditunjukkan di persamaan 2.3

$$S(a,b) = \begin{cases} 1 & a = b \\ 0 & a \neq b \end{cases} \quad (2.3)$$

Dimana:

a = kelas a pada training

b = kelas lain selain kelas a pada training

4. Menghitung *Weight voting*

Pada metode MKNN,langkah pertama adalah menhitung weighted masing masing tetangga. Lalu validitas dari setiap data pada data training dikalikan dengan *weighted* berdasarkan jarak Euclidean nya. Hasil dari tahap *weighted voting* ini berguna untuk mengetahui label dari kelas data yang diuji. Untuk menghitung *weight voting* dapat dilihat pada persamaan 2.4

$$W(i) = Validity(i)\ x \frac{1}{de+\alpha} \quad (2.4)$$

Dimana:

$W(i)$ = Perhitungan *weight voting*

$Validity(i)$ = Nilai validitas

de = Jarak Euclidean

Algoritma MKNN dirasa lebih baik dan akurat di banding metode KNN yang haya berdasarkan pada jarak. Dengan teknik *weighted voting* yang memiliki siknifikansi serta nilai validasi yang tinggi dan paling dekat dengan data bisa mengatasi kelemahan pada tiap data *error* yang memiliki masalah jarak dengan *outlier*. (Parvin, et al., 2008)

2.7 BM25

Metode *Best Matching* 25 merupakan metode untuk mencari peringkat oleh mesin pencari untuk memilah peringkat dokumen yang relevan terhadap permintaan pencarian yang ingin didapatkan. Di dalam metode ini terdapat skor, yang mana skor adalah merupakan suatu kombinasi skor linier weighted yang keseluruhan datanya merupakan dokumen uji. Di dalam dokumen uji tersebut, ada 3 hal yang dapat mempengaruhi bobot uji term ada dokumen, hal pertama adalah seberapa sering pencarian istilah itu muncul pada dokumen yang di sebut TF (Term Frequency). Hal kedua *inverse document frequency* (*IDF*) yang mana IDF ini terdapat pada dalam term tersebut, dan hal yang ketiga merupakan panjang dari dokumen. Analoginya dari panjang dokumen ini dapat mempengaruhi kualitas dokumen uji, jika suatu dokumennnya yang di cari semakin panjang dan tidak di ketahui, maka semakin banyak pula akan menyinggung kata yang berkaitan, namun jika semakin pendek, maka dokumen kata yang disinggung jumlahnya juga akan semakin sedikit. Dengan kata lain dari 3 hal tersebut merupakan faktor utama yang memengaruhi perhitungan dari metode BM25 (Russel dan Norvig, 2012). Perhitungan pada BM25 dapat dilihat pada persamaan 2.5.

$$BM25 = \sum_{i=1}^{|q|} idf(q_i).\frac{tf(q_i,d).(k_1+1)}{tf(q_i,d)+k_i.(1-b+b.\frac{|d|}{dl_{avg}})} \quad (2.5)$$

Keterangan:

$idf(qi)$: nilai *invers document frequency* pada *term query i*

$tf(qi,d)$: jumlah frekuensi *term query i* pada dokumen *j*

$k1$: 1,2 ≤ $k1$≤ 2,0

b : 0,5 ≤ b ≤ 0,8

$dlavg$: rata-rata panjang semua dokumen

$|d|$: panjang dokumen

Dengan persamaan *inverse document frequency* yang ditunjukkan pada Persamaan 2.6

$$idf(qi) = log\left(\frac{N - df(qi) + 0{,}5}{df(qi) + 0{,}5}\right) \quad (2.6)$$

Keterangan:

$idf(qi)$: nilai *invers document frequency* pada *term query i*

N : total dokumen dalam koleksi

$df(qi)$: jumlah dokumen yang terdapat *term query i*

2.8 Evaluasi Hasil

Untuk mengukur seberapa baik metode klasifikasi bekerja maka perlu dilakukan evaluasi. Salah satu metode untuk mengukur hasil dari kinerja suatu sistem adalah *Confusion Matrix.* Pada dasarnya confusion matrix berisi informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan hasil yang harus diklasifikasikan. Evaluasi yang dilakukan yaitu dengan cara mencari nilai *precision, recall*, dan akurasi. Penggunaan *Confusion Matrix* dapat dilihat pada tabel 2.2.

Tabel 2.2 *Confusion Matrix*

Kelas	Prediksi Positif	Prediksi Negatif
Nilai Sebenarnya Positif	TP	FN
Nilai Sebenarnya Negatif	FP	TN

Keterangan:

TP = jumlah contoh positif yang diklasifikasikan secara benar (*True Positive*)

TN = jumlah contoh negatif yang diklasifikasikan secara benar (*True Negative*)

FP = jumlah contoh negatif yang salah diklasifikasikan sebagai positif (*False Positive*)

FN = jumlah contoh positif yang salah diklasifikasikan sebagai negatif (*False Negative*)

Perhitungan *precision, recall*, dan akurasi dapat dilihat pada persamaan 2.7, persamaan 2.8, dan persamaan 2.9

$$Precision_i = \frac{\text{TP}}{\text{FP} + TP_i} \quad (2.7)$$

$$Recall_i = \frac{TP}{FN+TP} \quad (2.8)$$

$$\text{Akurasi} = \frac{TP + TN}{TP+FP+TN+FN} \quad (2.9)$$

BAB 3
METODOLOGI

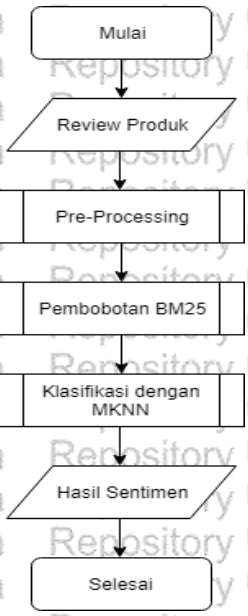
Pada bab metodologi ini menunjukkan secara terperinci mengenai urutan prpses yang akan dilakukan pada penelitian ini. Proses yang akan dijelaskan yaitu tipe penelitian yang digunakan, strategi penelitian berupa studi kasus, partisipan penelitian, lokasi yang digunakan untuk penelitian, data penelitian, dan implementasi algoritme.

3.1 Tipe Penelitian

Tipe penelitian pada penelitian ini merupakan non implementatif analitik. Penelitian non-implementatif yaitu memfokuskan pada fenomena tertentu sehingga kemudian akan di hasilkan tinjauan ilmiah. Penelitian analitik memfokuskan pada hasil analisis yang kemudian akan di bandingkan dan selanjutnya dijadikan bahan perbaikan atau.

3.2 Strategi Penelitian

Untuk mengembangkan penelitian ini, strategi yang akan dilakukan pertama adalah dengan mengamati permasalahan yang ada di sekitar dan didukung dengan dari referensi pada penelitian – penelitian yang sudah dilakukan yang selanjutnya permasalahan tersebut akan dijadikan objek penelitian. Didalam penelitian ini objek yang digunakan adalah *review* produk dari aplikasi Sociolla. Setelah menemukan permasalahan nya, akan dicari referensi pendukung dari penelitian terdahulu untuk membantu penelitian ini. Alur proses nya dapat dilihat pada gambar 3.1



Gambar 3.1 Alur Proses Sistem

3.3 Partisipan Penelitian

Partisipan yang terlibat didalam penelitian ini adalah konsumen pada aplikasi Sociolla berdasarkan *review* yang ada. Partisipan tersebut dipilih untuk mengetahui permasalahan dan sebagai validitas data saja. Selain itu dalam penelitian terdapat pakar yang nantinya akan memberikan label pada tiap dokumen.

3.4 Lokasi Penelitian

Lokasi penelitian ini dilakukan di Laboratorium Riset Fakultas Ilmu Komputer (FILKOM) Universitas Brawijaya.

3.5 Teknik Pengumpulan Data

Pengumpulan data yang dilakukan pada penelitian ini menggunakan data primer yaitu didapatkan langsung dari *review* produk pada aplikasi Sociolla dan juga dapat dibuka melalui portal web https://*review*.soco.id

3.6 Data Penelitian

Data yang dipakai berasal dari *review* produk Nature Republic Aloe Vera Soothing Gel yang ada di aplikasi Sociolla. Data tersebut 500 data, termasuk 250 data positif dan 250 data negatif

3.7 Teknik Analisis Data

Teknik analisis data dilakukan melalui proses perbandingan hasil penelitian dan hasil pengujian. Hasil prediksi analisis sentimen yang telah diperoleh dibandingkan dengan hasil dari pemberian label secara manual. Pengujian yang akan dilakukan menggunakan *confusion matrix* yang Selanjutnya, dilakukan perhitungan akurasi,presisi,*recall,*dan *f-measure*.

3.8 Implementasi Algoritme

Tahap implementasi algoritme menggunakan *Modified KNN* dan BM25 yang pertama dilakukan pengolahan data kemudian dilakukan perhitungan manualisasi untuk membandingkan antara hasil perhitungan manualisasi maupun dengan pengujian metode pada penelitian ini dan juga untuk mempermudah peneliti mengetahui langkah perhitungan. Tahap selanjutnya dengan proses pembagian data latih serta data uji, lalu untuk melakukan evaluasi memakai akurasi.

BAB 4 PERANCANGAN

Pada bab perancangan ini, berisi tentang bentuk perancangan sistem yang akan diterapkan sehingga dapat mengklasifikasikan suatu opini berupa *review* produk. Pada bab ini juga akan dibahas terkait manualisasi atau perhitungan manual selain itu juga dibahas mengeani perancangan pengujian.

4.1 Deskripsi Permasalahan

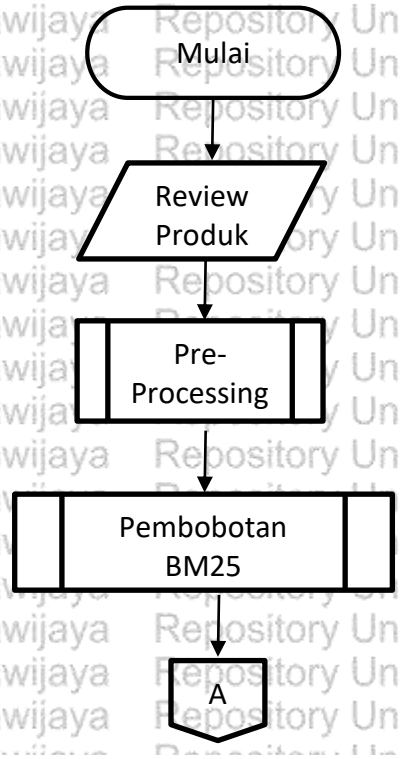
Dengan adanya perkembangan digital, semua kegiatan manusia menjadi lebih mudah termasuk ketika akan berbelanja. Hal ini ditunjukkan dengan berkembangnya *e-commerce* yaitu sebuah penjualan, pembelian dan penyebaran berupa barang atau jasa berbasis sistem elektronik. Berbagai kemudahan dirasakan dengan adanya *e-commerce* seperti memudahkan komunikasi antara produsen dan konsumen, produsen tidak perlu mendirikan toko yang besar untuk menjual barang atau jasanya, transaksi pembayaran yang dilakukan juga lebih efisien karena dapat dilakukan secara *online* dan kebanyakan *e-commerce* yang ada tersedia 24 jam.

Salah satu *e-commerce* yang bergerak di industri kecantikan adalah PT Social Bella Indonesia atau yang lebih dikenal dengan Sociolla. Perusahaan ini berdiri sejak tahun 2015 dan terus berkembang pesat. Hingga saat ini Sociolla bukan sekedar *e-commerce* yang hadir dalam bentuk *online*, tetapi juga *offline*. Bukan hanya itu bisnis lainnya yang mereka kerjakan berupa sebuah media bernama *Beauty Journal*. Media ini pada awalnya hanya sebuah blog, tetapi lama kelamaan berkembang menjadi marketing agency yang berisi berbagai informasi sehingga dapat membantu mengedukasi konsumen salah satunya juga terdapat review dari konsumen yang telah menggunakan produk tertentu. Dengan adanya review ini pastinya sangat membantu konsumen yang hendak membeli produk kecantikan dengan membaca terlebih dahulu review yang ada, tetapi seringkali opini atau review yang ada sangatlah banyak bahkan bisa sampai ribuan. Tak heran jika konsumen juga terkadang kesulitan menentukan apakah produk tersebut baik atau tidak. Maka dari itu penulis ingin menyelesaikan permasalahan ini dengan cara mengklasifikasi yang berasal dari review yang ada pada produk di Sociolla menggunakan algoritme BM25 dan MKNN.

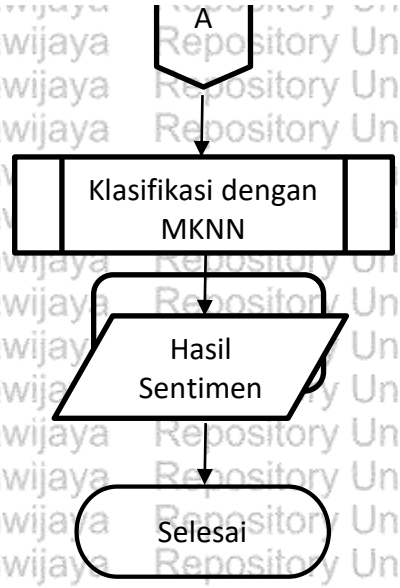
4.2 Deksripsi Umum Sistem

Klasifikasi Review Produk Kecantikan Pada Aplikasi Sociolla Menggunakan Algoritme Modified K-Nearest Neighbor (MK-NN) dengan Pembobotan BM25 merupakan sebuah sistem yang dikembangkan untuk mengklasifikasikan ulasan atau opini berupa *review* dari produk yang ada pada aplikasi Sociolla yang mengandung opini positif dan negatif. Data yang digunakan data dari aplikasi Sociolla yang berisi ulasan, yang telah diambil yang terdiri dari data latih dan data uji.

Untuk dapat mengklasifikasikan *review* ke dalam kelas positif atau negative, diawali dengan melakukan *pre-processing* ke dalam data latih dan data uji yang berguna untuk pembersihan data. Tahapan dari *pre-processing* yaitu *Case Folding, Tokenisasi, Filtering,* dan *Stemming.* Lalu langkah selanjutnya dengan melakukan pembobotan pada tiap *term*nya sehingga dapat dihitung kemiripannya. Pembobotan yang digunakan adalah BM25. Setelah didapatkan nilai bobot dari BM25, lalu dihitung validitas data, kemudian data uji yang telah ada dihitung kemiripan nya menggunakan BM25 juga, lalu dilakukan *weight voting* kemudian akan didapatkan hasil klasifikasi menggunakan metode *Modified KNN.* Diagram alir gambaran sistem dapat dilihat pada Gambar 4.1



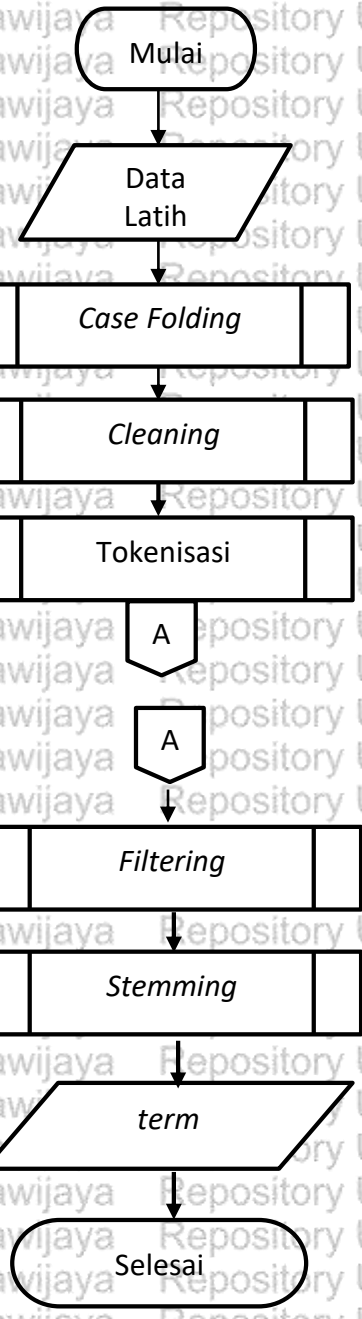
Gambar 4.1 Diagram Alir Gambaran Umum Sistem



Gambar 4.2 Diagram Alir Gambaran Umum Sistem (lanjutan)

4.3 Alur *Pre-Processing*

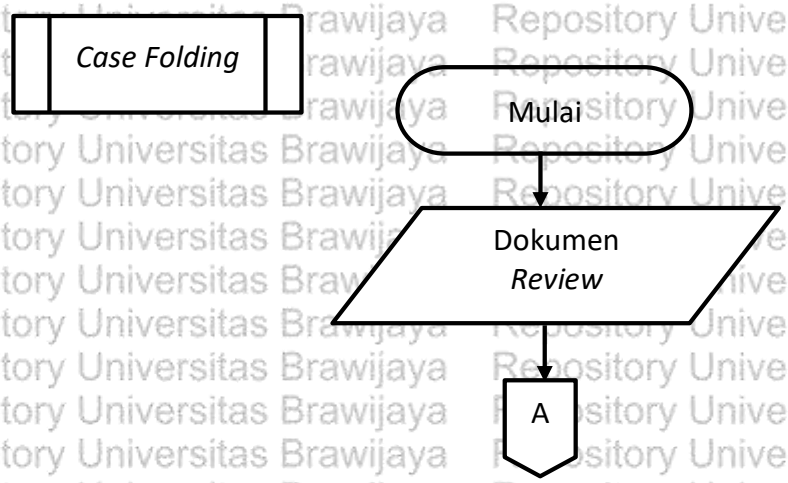
Step pertama adalah melakukan *pre-processing*. Pada Gambar 4.2 tersebut menjelaskan alur dari tahapan *pre-processing*. Pertama dokumen yang dimasukkan sudah ditentukan kelasnya. Kemudian sistem melakukan *looping* sebanyak dokumen yang ada. Setelah itu masuk ke tahap *pre-processing*, langkah-langkah nya yaitu *Case Folding*, *Cleaning*, Tokenisasi, *Filtering* dan juga *Stemming*. Diagram alir proses *pre-processing* dapat dilihat pada Gambar4.3



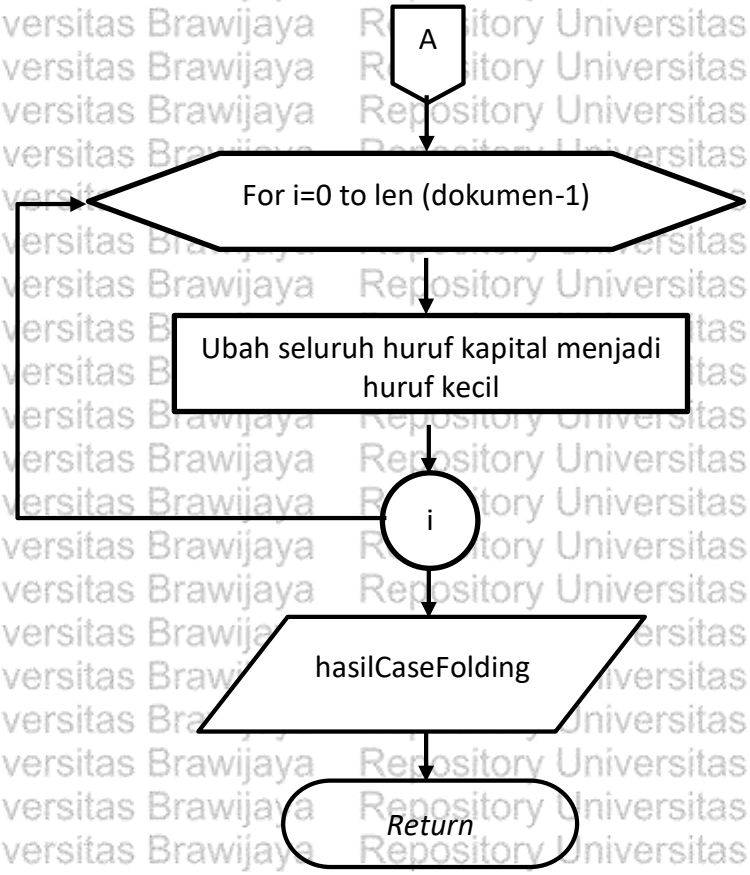
Gambar 4.3 Diagram Alir Proses *Pre-Processing*

4.3.1 Case Folding

Case folding merupakan langkah pertama dalam proses *preprocessing* yaitu mengubah seluruh huruf capital menjadi *lower case* atau huruf kecil, sehingga tiap kata dalam dokumen sama. Proses *case folding* dapat dilihat pada Gambar 4.4.



Gambar 4.4 Diagram Alir *Case Folding*

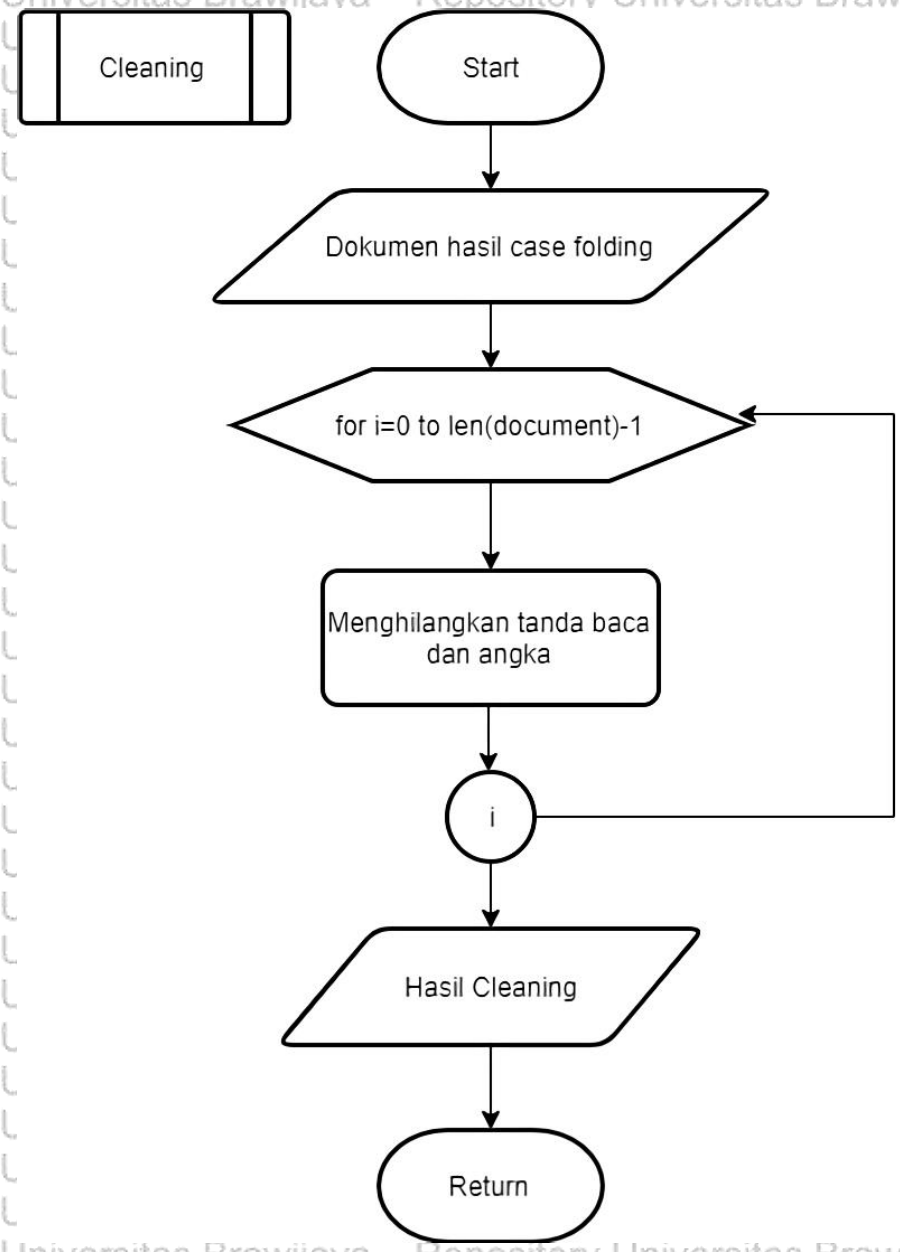


Gambar 4.5 Diagram *Alir Case Folding (lanjutan)*

Pada gambar 4.44 terdapat perulangan pada variable i hingga panjang dokumen jika terdapat huruf kapital maka akan diubah menjadi *lowercase*, lalu *looping* akan terus dilakukan apabila masih terdapat huruf besar maka akan didapatkan keluaran berupa hasil case *folding* berupa huruf kecil semua.

4.3.2 Cleaning

Tahap selanjutnya pada proses pre-processing adalah *cleaning*. *Cleaning* adalah tahapan untuk membersihkan angka, tanda baca dan karakter lainnya yang bukan huruf. Sehingga semua dokumen berisi huruf saja. Proses *cleaning* dapat dilihat pada Gambar 4.6.

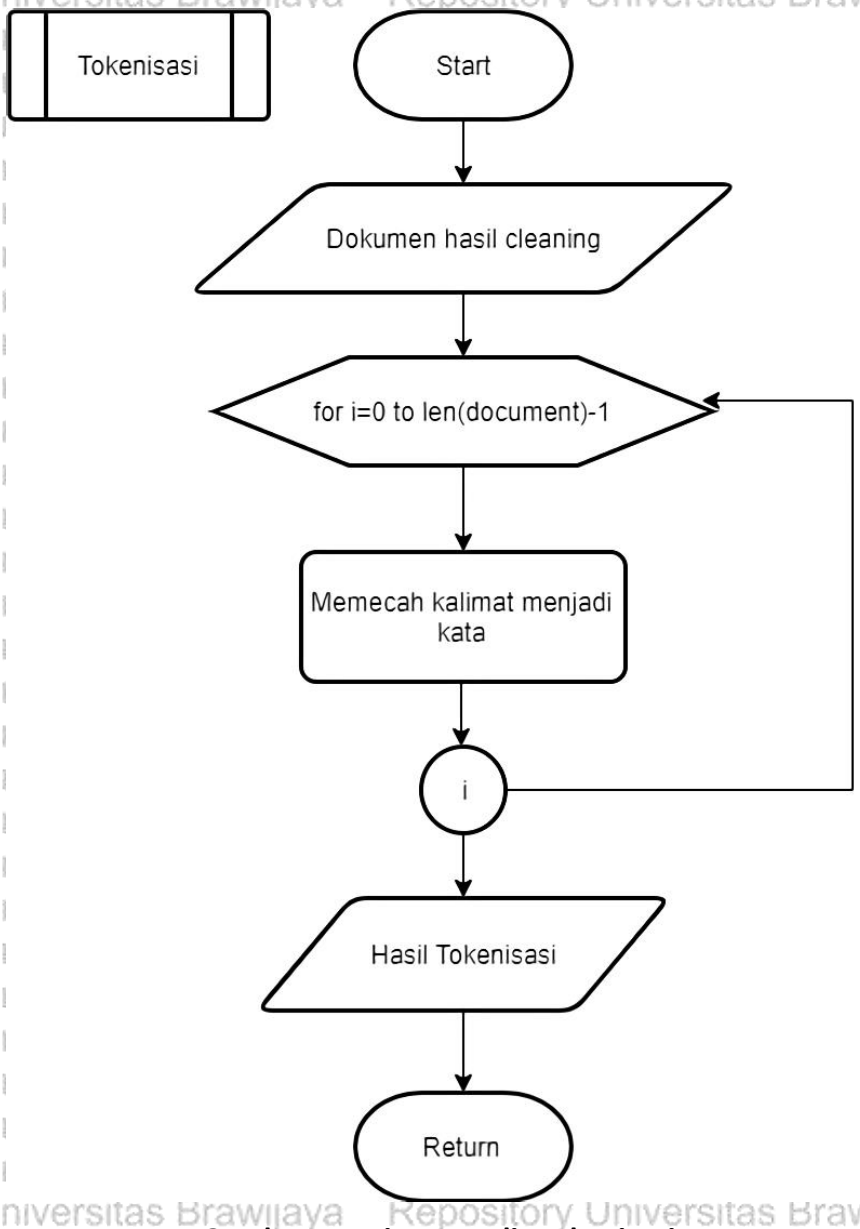


Gambar 4.6 Diagram Alir *Cleaning*

Berdasarkan gambar 4.6 data yang telah melalui tahapan *case folding* akan dilakukan *cleaning* yaitu melakukan perulangan pada dokumen jika terdapat tanda baca atau angka maka akan dihilangkan sehingga menghasikjan *output* berupa hasil *cleaning.*

4.3.3 Tokenisasi

Tahap selanjutnya pada *pre-processing* yaitu tokenisasi. Tokenisasi adalah proses memisah dokumen berupa kalimat menjadi kata-kata Gambar 4.7 menunjukkan tahapan dari tokenisasi.

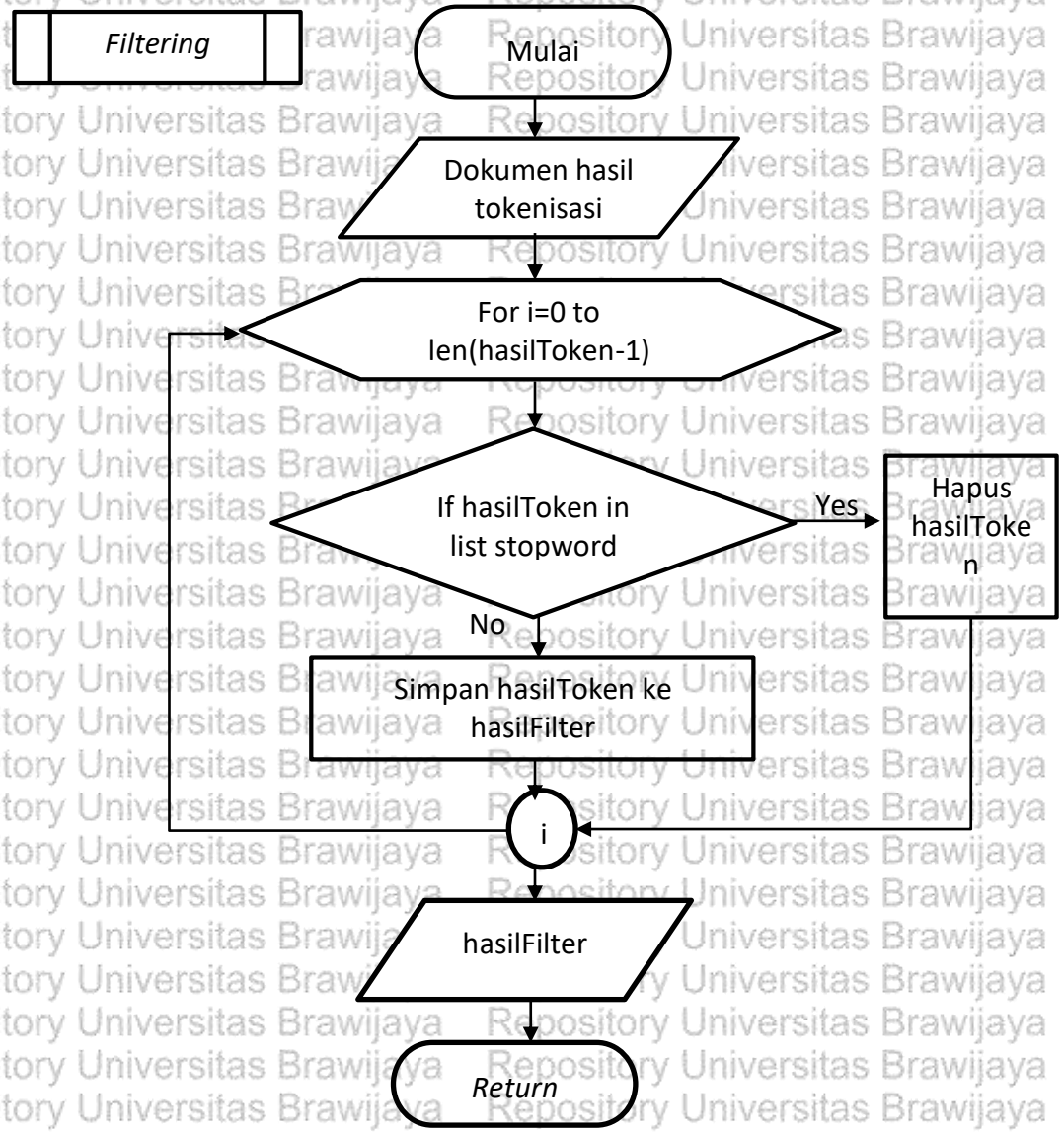


Gambar 4.7 Diagram Alir Tokenisasi

Dokumen yang telah dilakukan *case folding dan cleaning* akan masuk kedalam proses tokenisasi. Kemudian dilakukan *looping* sebanyak jumlah dokumen, kemudian dipecah menjadi kata-kata.

4.3.4 Filtering

Filtering merupakan tahap selanjutnya yaitu membuang term yang tidak mewakili dokumen yang mana term tersebut tidak relevan dengan dokumen. Untuk membuang term ini digunakan *stoplist* atau s*topword* yaitu berisi kata-kata yangtidak penting yang tidak digunakan. *Stoplist* yang digunakan dalam sistem ini adalah *stoplist* Tala. Tahap *filtering* dapat dilihat pada Gambar 4.8.

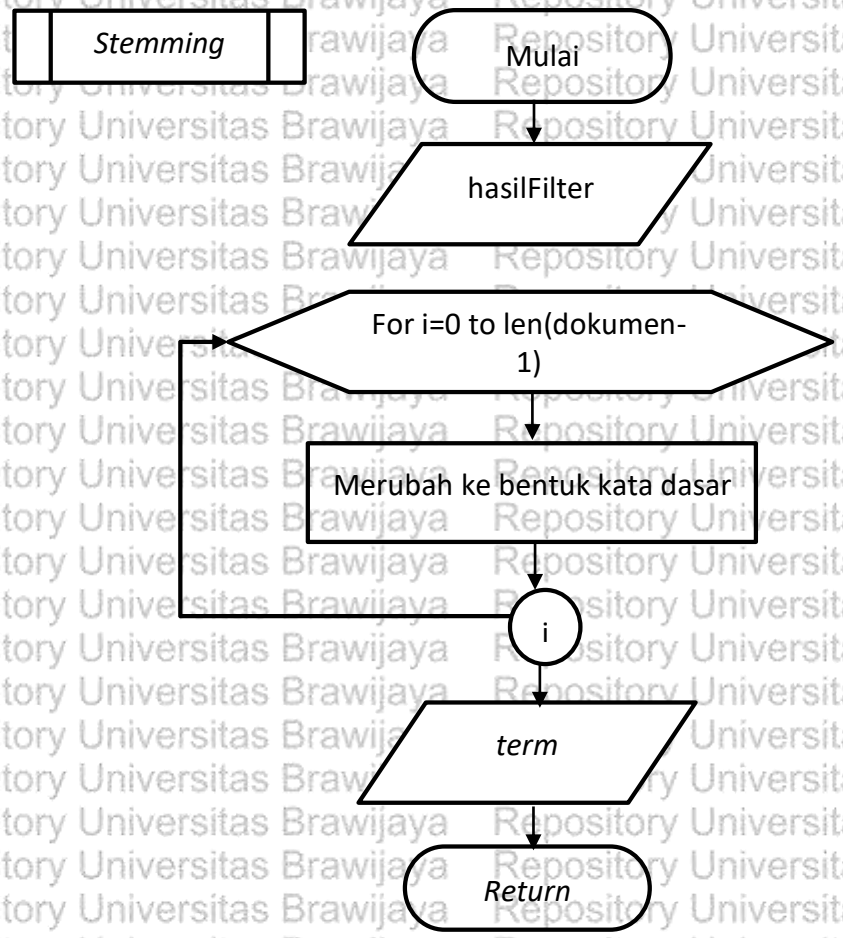


Gambar 4.8 Diagram Alir *Filtering*

Hasil dari proses tokenisasi digunakan untuk tahap selanjutnya yaitu tahap *filtering* . Pertama sistem akan membaca *stoplist*, kemudian dilakukan *looping* sebanyak dokumen, lalu dicocokkan dengan *stoplist*. Apabila ada term yang sama pada stoplist maka term itu akan dihapus. Sebaliknya jika ada term yang tidak ada di stoplist maka term tersebut yang akan digunakan untuk tahapan selanjutnya.

4.3.5 Stemming

Stemming adalah proses terakhir pada *pre-processing*. Pada tahap ini dokumen yang digunakan adalah hasil dari proses *filtering.* Dokumen yang ada pada data latih yang terdapat imbuhan akan diubah menjadi kata dasar seperti pada Gambar 4.9.

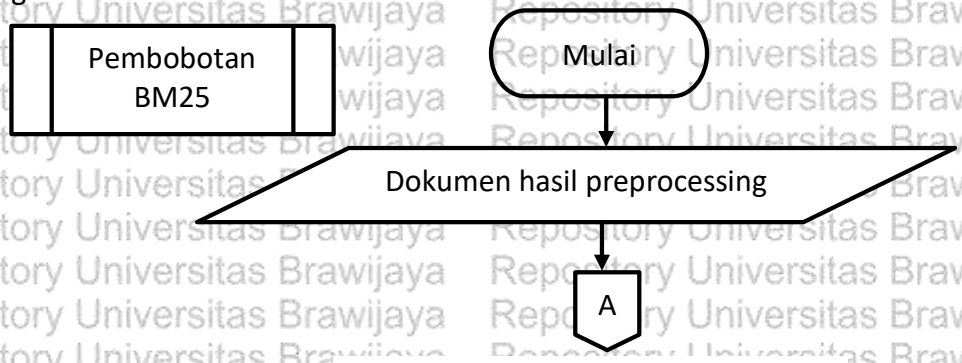


Gambar 4.9 Diagram Alir *Stemming*

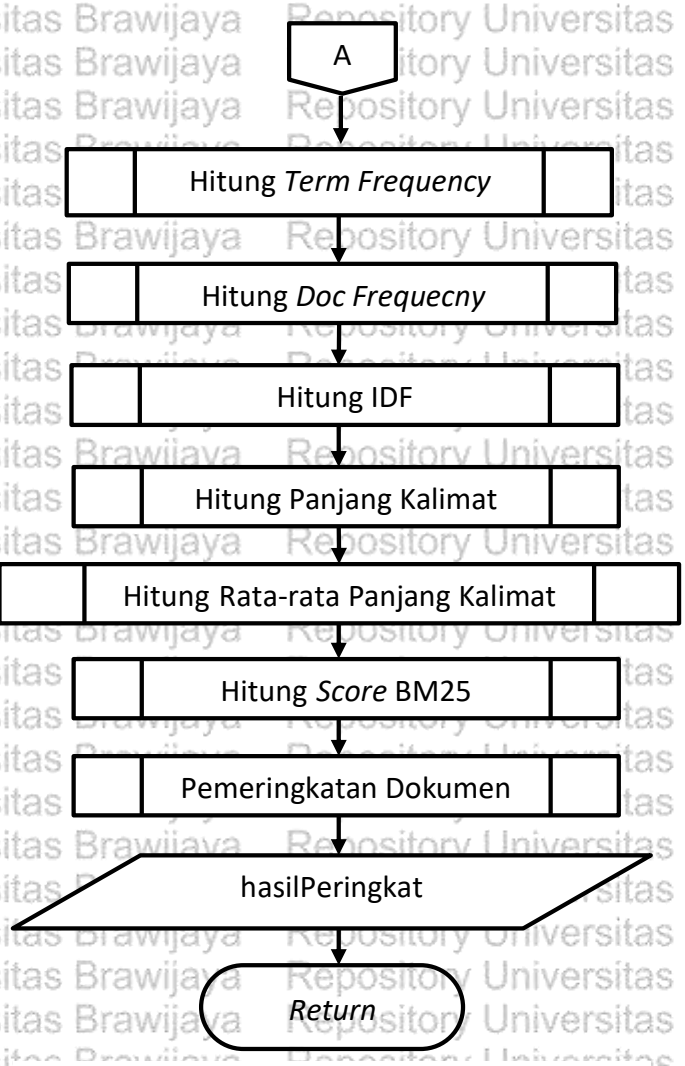
Pada gambar 4.9 data hasil filtering akan dilakukan proses stemming, terdapat perulangan pada proses ini jika panjang dokumen I masih terdapat kata yang memiliki imbuhan atau bukan kata dasar, pada tahap ini digunakan *library* Sastrawi.

4.4 Alur Proses BM25

Tahapan pada proses pembobotan menggunakan BM25 dapat dilihat pada gambar 4.10



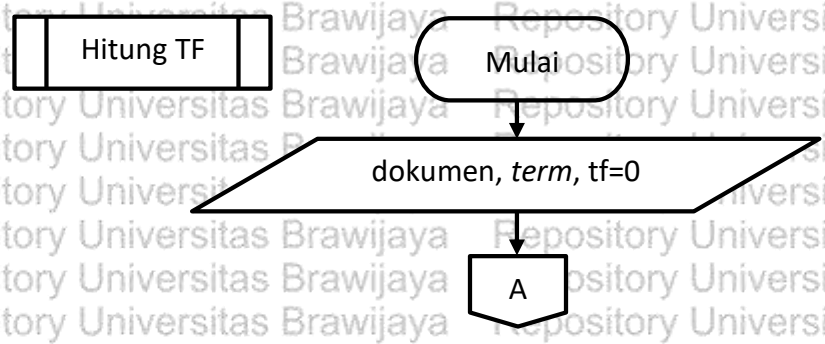
Gambar 4.10 Diagram Alir BM25



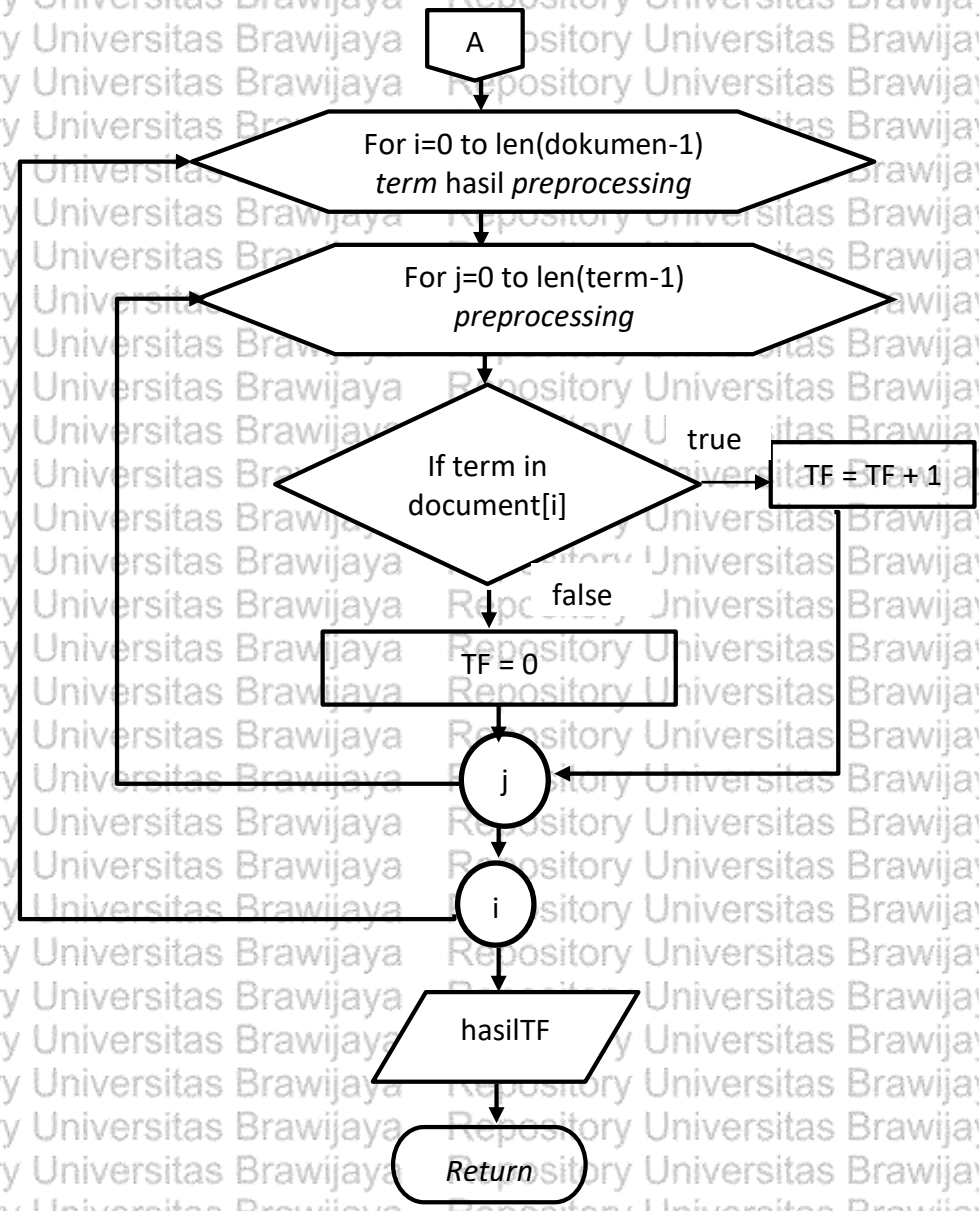
Gambar 4.11 Diagram Alir BM25 (lanjutan)

4.4.1 Hitung *Term Frequency*

Term Frequency adalah jumlah kata yang muncul pada dokumen. Setelah tahap *pre-processing* kemudian masuk ke perhitungan *term frequency*. Alur proses perhitungan TF dapat dilihat pada Gambar 4.12.



Gambar 4.12 Diagram Alir TF

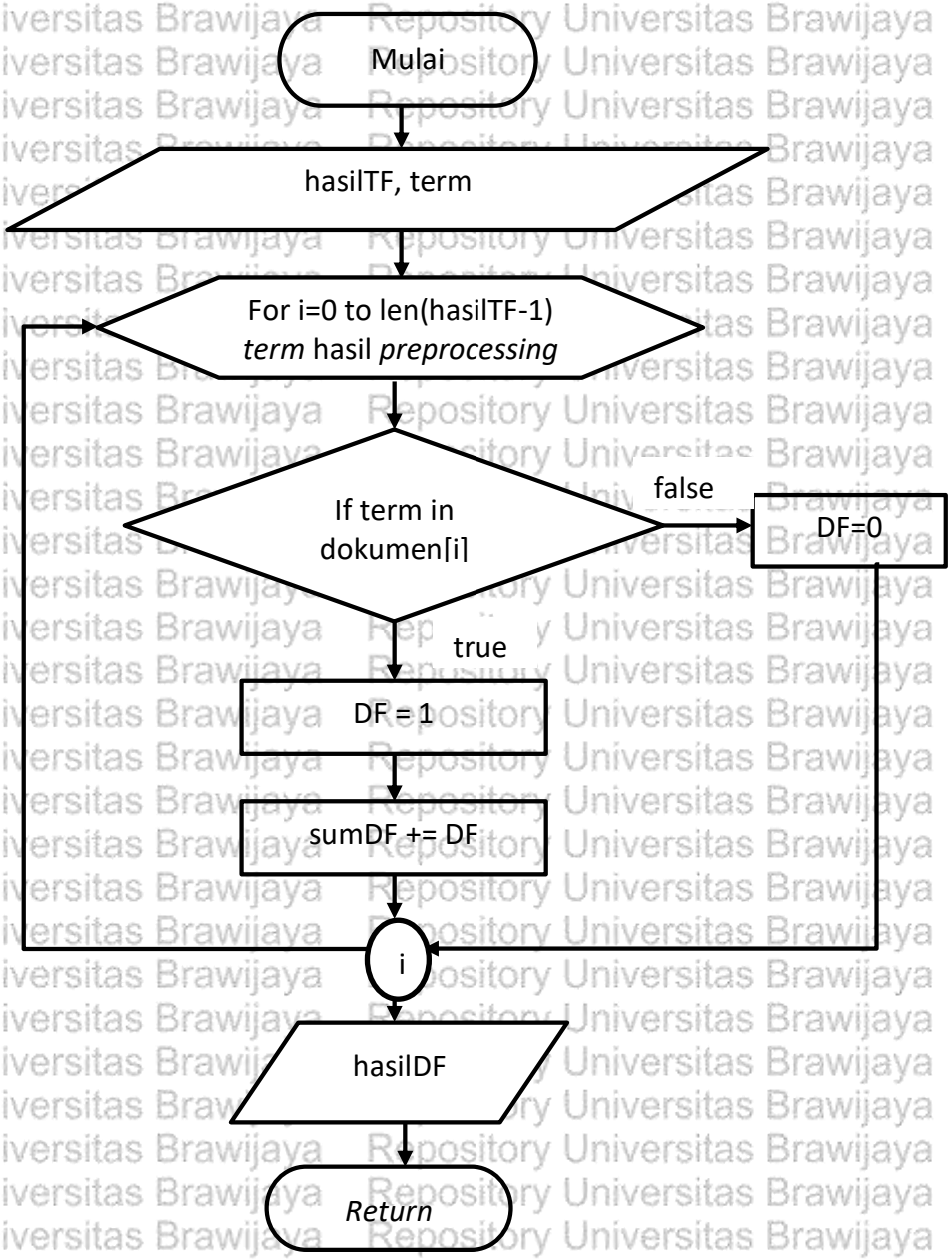


Gambar 4.13 Diagram Alir TF (lanjutan)

Pertama sistem akan melakukan *looping* sebanyak total dokumen, kemudian perulangan sebanyak indeks *term* yang ada pada dokumen latih. Lalu dilakukan seleksi kondisi yang berguna untuk mencocokkan data uji dengan *term* yang terdapat di data latih. Apabila terdapat *term* dalam dokumen uji yang tidak cocok dengan dokumen yang ada di data latih maka nilainya 0, sedangkan Jika cocok, *term* dalam dokumen uji akan menambahkan jumlah kata yang sama dalam dokumen latih.

4.4.2 Hitung *Document Frequency* (DF)

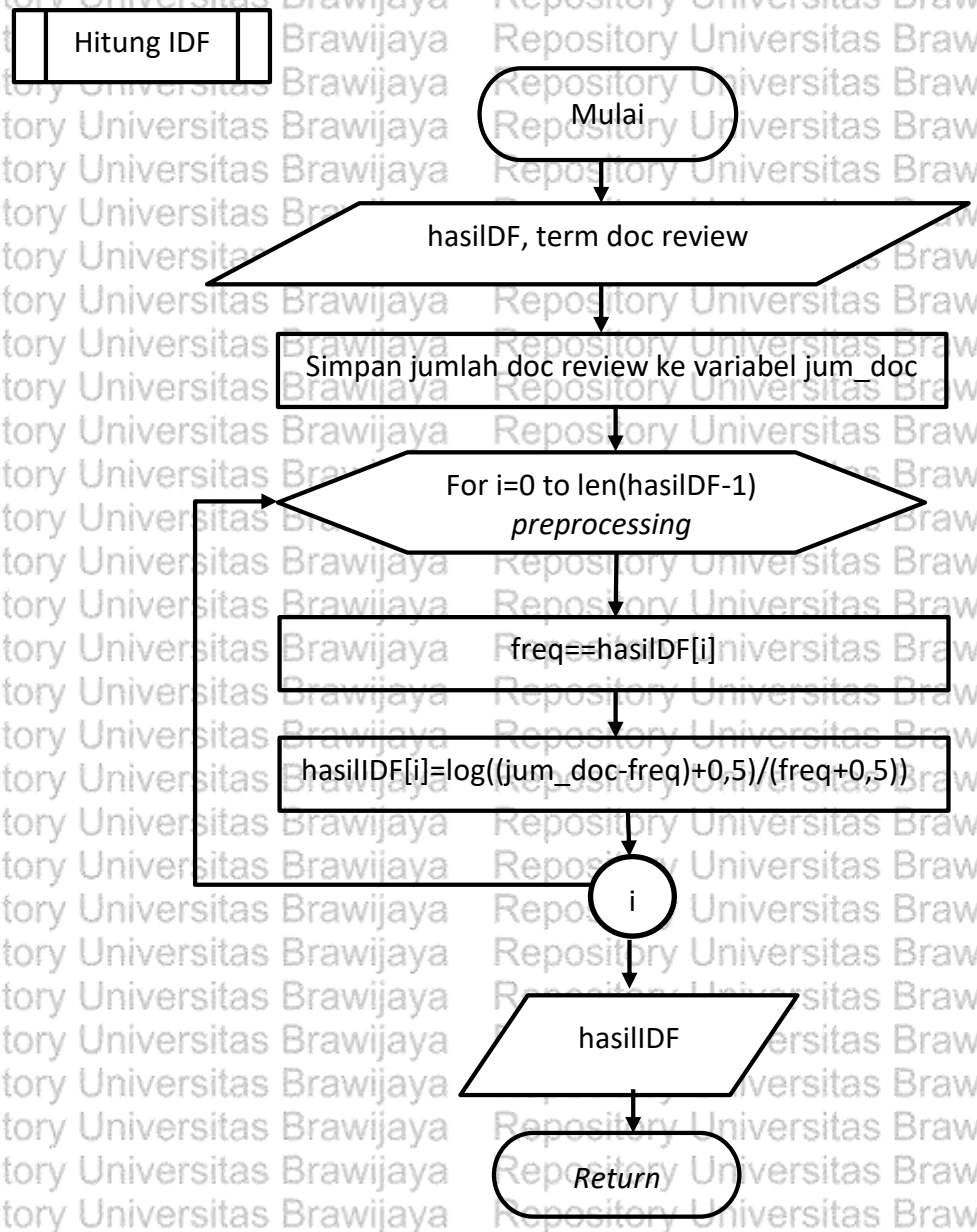
Setelah menghitung tf langkah selanjutnya adalah menghitung *Document frequency*. Untuk mendapatkan berapa jumlah dari suatu *term* pada setiap dokumen digunakan dengan menghitung banyaknya kemunculan *term* pada data uji. Proses perhitungan DF dapat dilihat pada Gambar 4.14.



Gambar 4.14 Diagram Alir DF

4.4.3 Hitung *Inverse Document Frequency* (IDF)

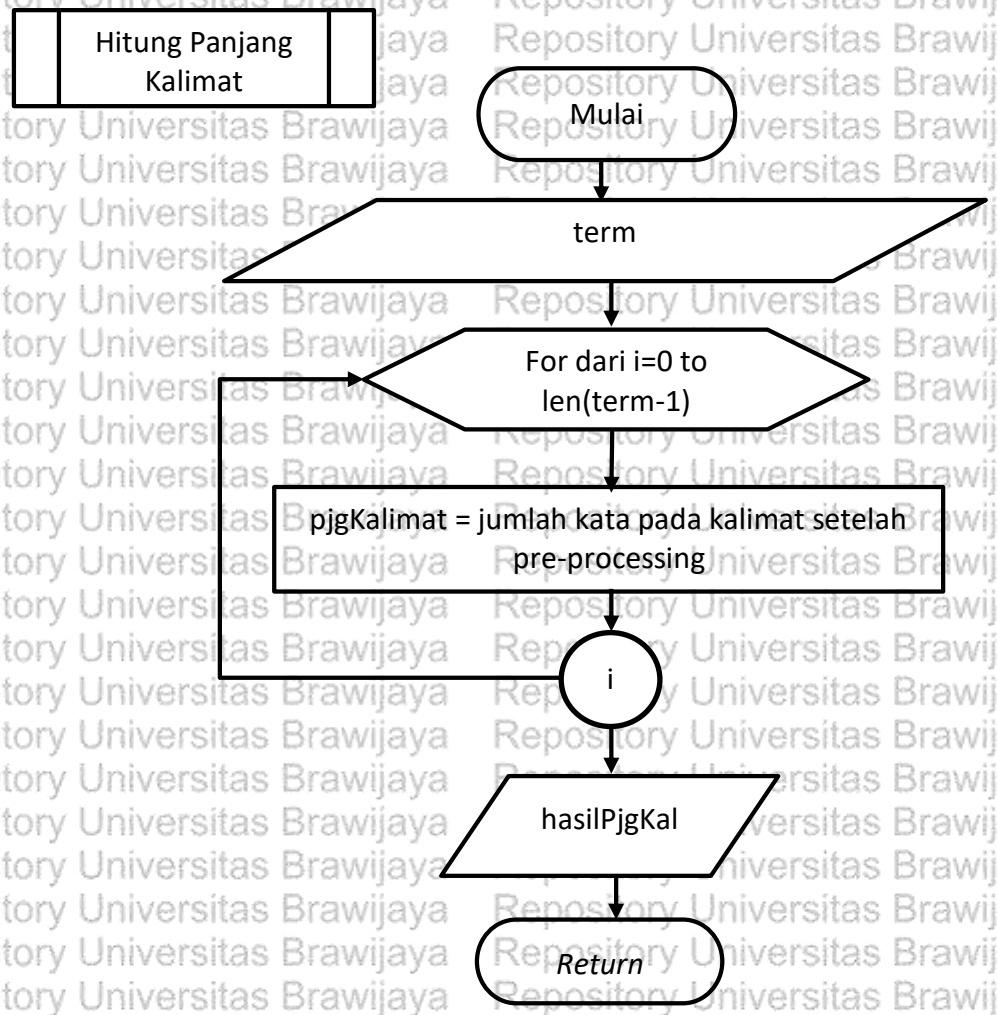
Setelah menghitung nilai *df* tahap selanjutnya adalah menghitung *idf*. *Idf* adalah nilai invers di semua total dokumen yg didalamnya terdapat term yang dicari. Tahap *idf* dapat dilihat pada Gambar 4.15.



Gambar 4.15 Diagram Alir IDF

4.4.4 Hitung Panjang Kalimat

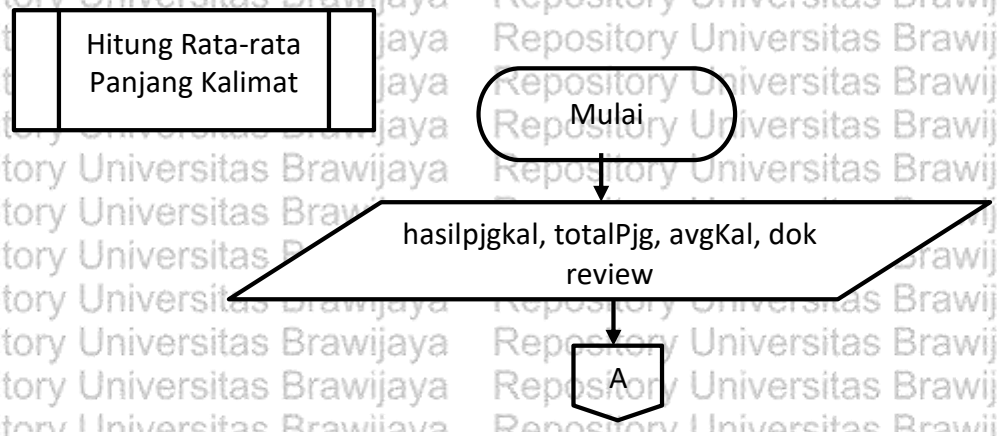
Setelah dilakukan perhitungan IDF langkah selanjutnya adalah menghitung panjang kalimat yaitu menghitung seluruh jumlah setiap kata pada dokumen. Tahapan untuk proses menghitun panjang kalimat terdapat pada Gambar 4.16.



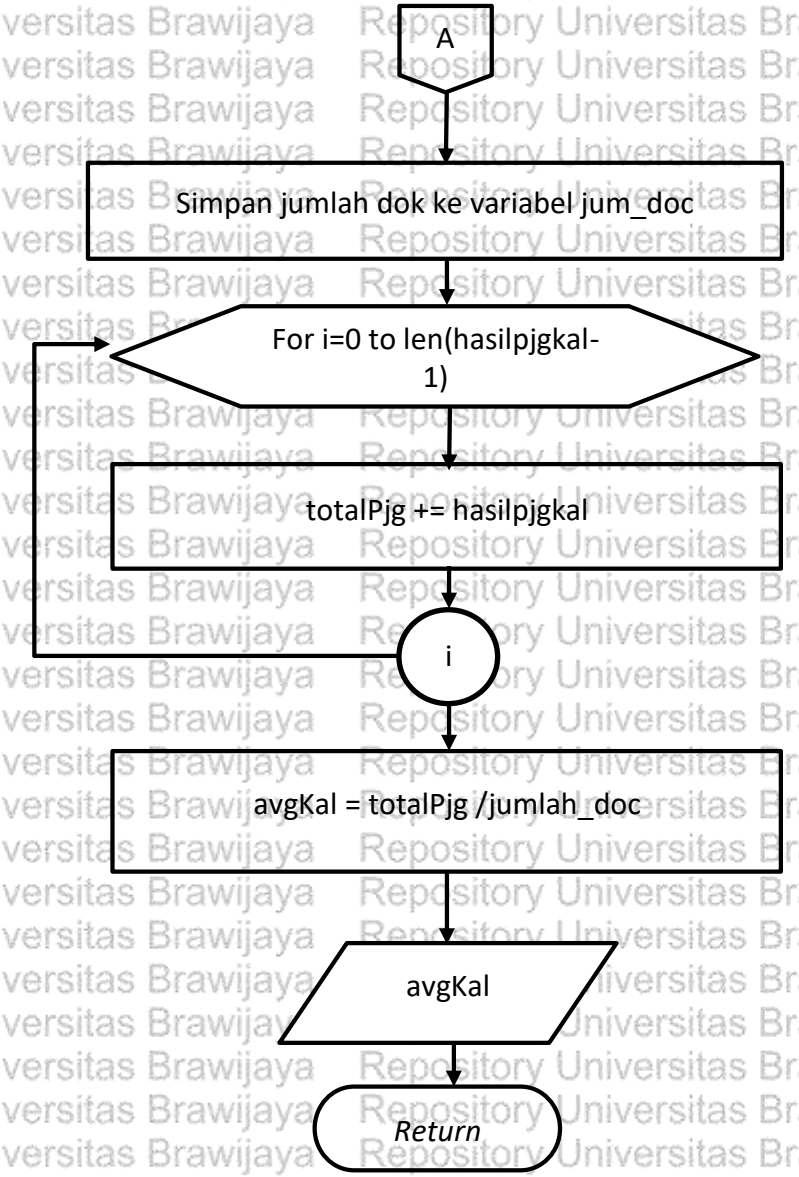
Gambar 4.16 Diagram Alir Hitung Panjang Kalimat

4.4.5 Hitung Rata-rata Panjang Kalimat

Setelah didapat panjang kalimatnya, selanjutnya dihitung rata-rata nya. Gambar 4.17 menunjukkan proses menhitung rata-rata panjang kalimat.



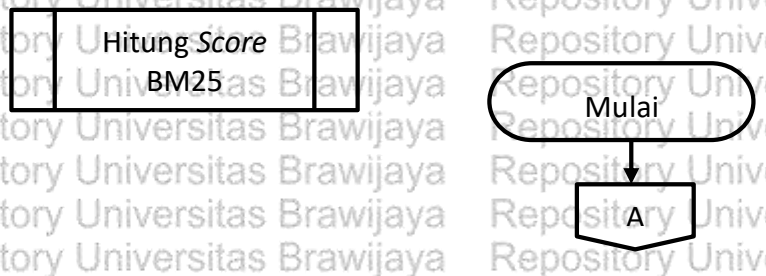
Gambar 4.17 Diagram Alir Rata-Rata Panjang Kalimat

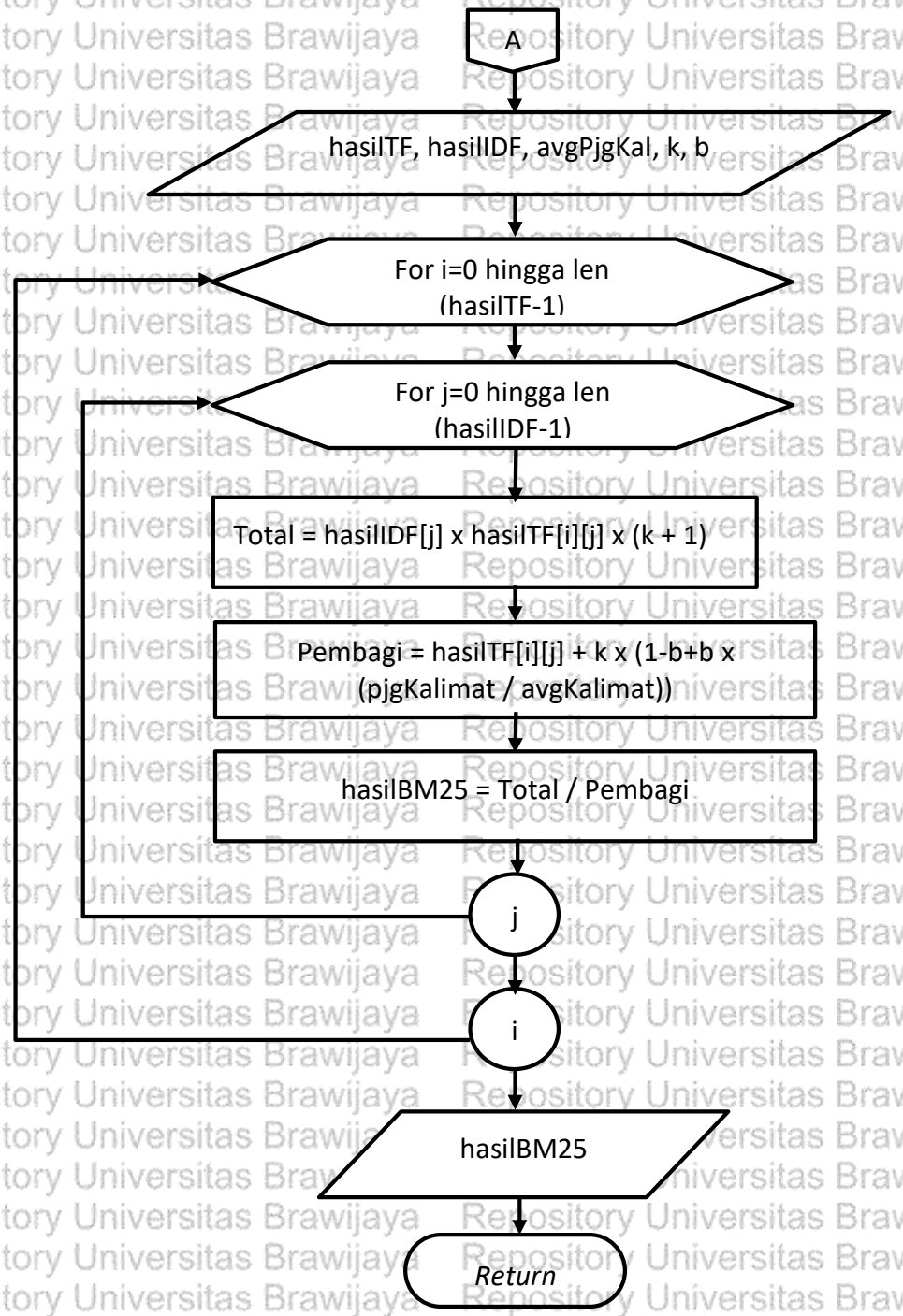


Gambar 4.18 Diagram Alir Rata-rata Panjang Kalimat (lanjutan)

4.4.6 Hitung *score* BM25

Setelah didapatkan nilai *idf* maka dilakukan perhitungan *score* dengan BM25. BM25 disini berguna untuk mencocokkan dokumen berdasarkan *term* yang dicari. Gambar 4.19 menunjukkan proses menghitung BM25.

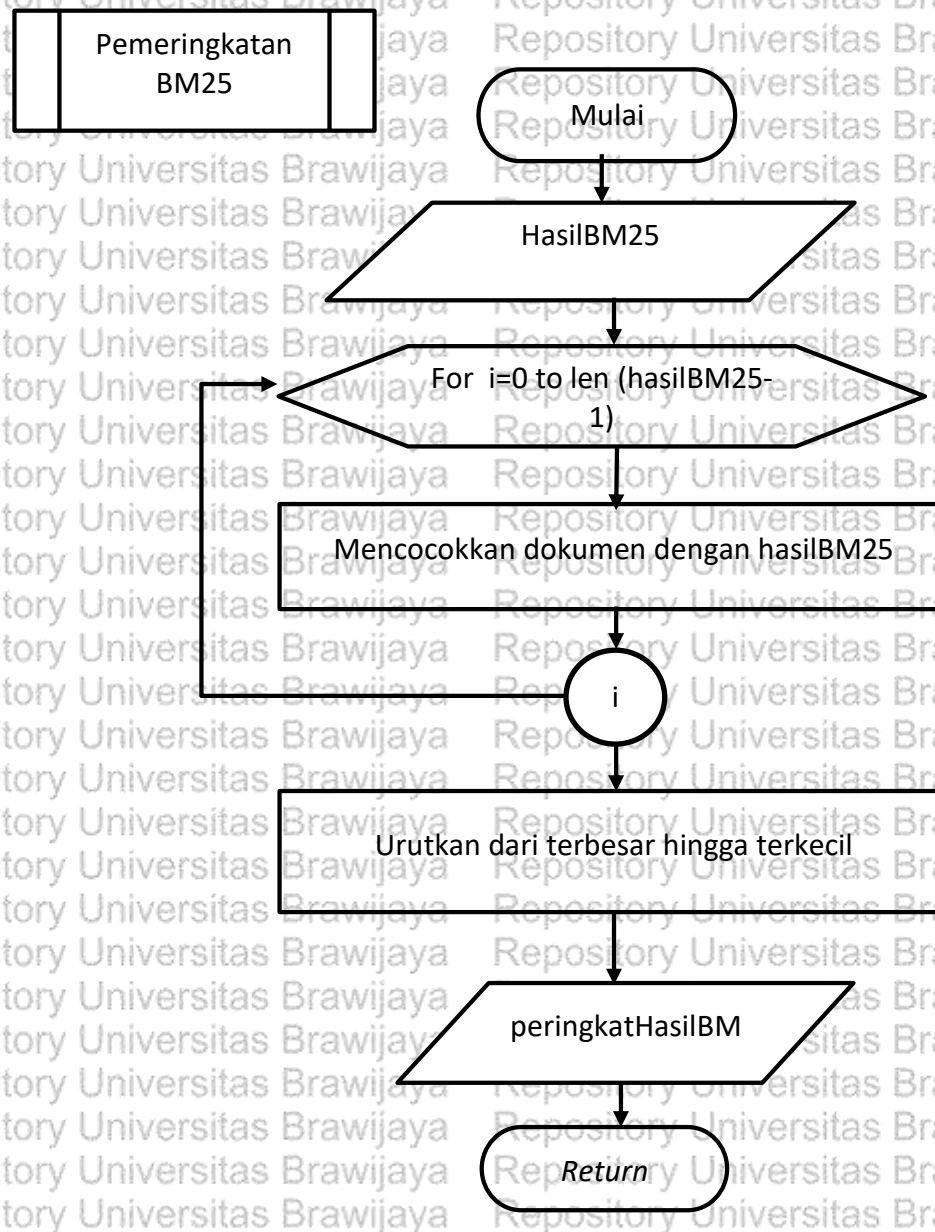




Gambar 4.19 Diagram Alir *Score* BM25

4.4.7 Hitung pemeringkatan BM25

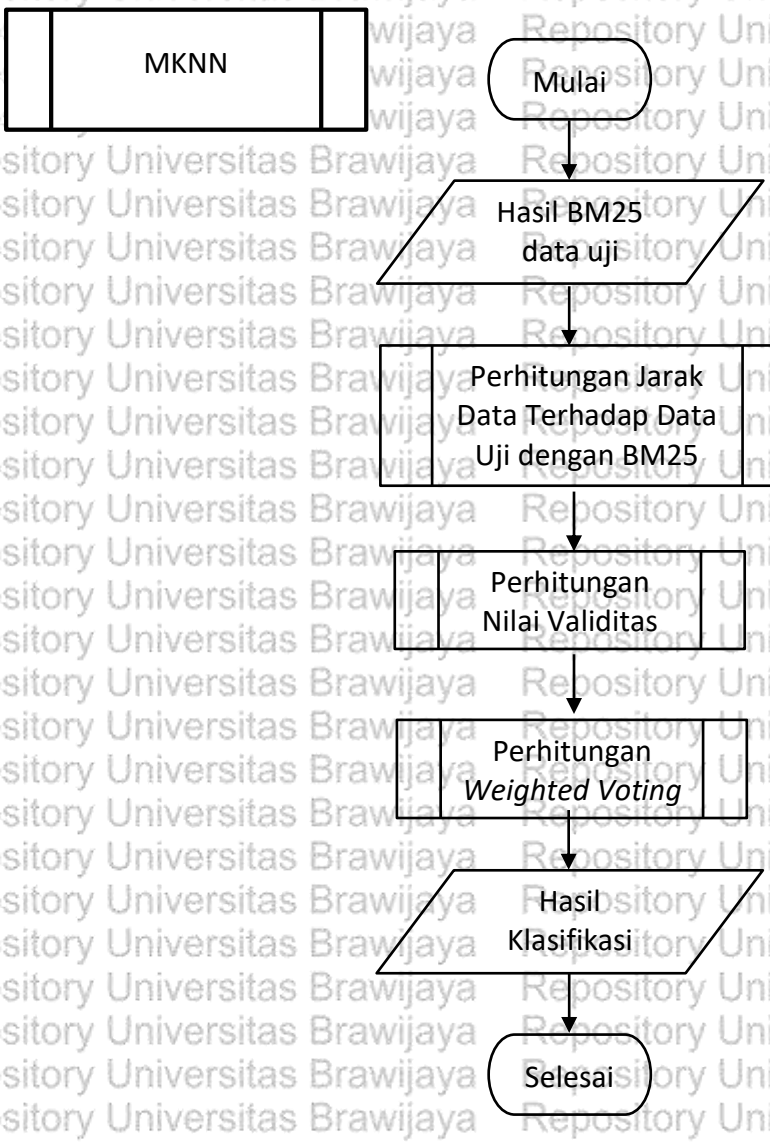
Tahap terakhir dari perhitungan BM25 adalah mengurutkan hasil *score* yang didapat dari nilai paling besar hingga paling kecil. Gambar 4.20 menunjukkan proses pemeringkatan BM25.



Gambar 4.20 Diagram Alir Pemeringkatan BM25

4.5 Alur Proses MKNN

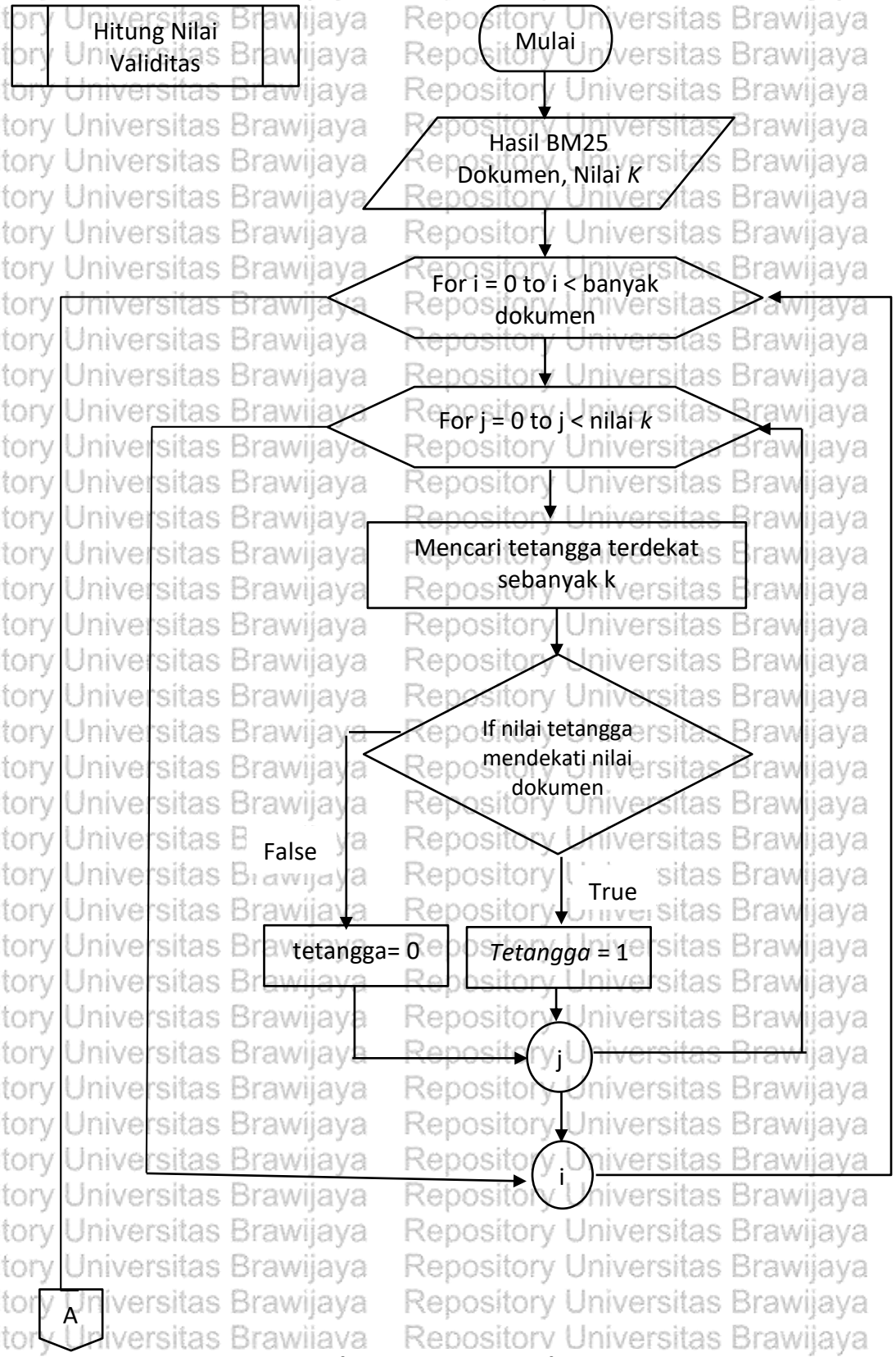
Setelah dilakukan serangkaian proses untuk pembobotan dengan metode BM25 langkah selanjutnya adalah tahap klasifikasi dengan metode MKNN. Gambar 4.21 menunjukkan alur proses MKNN.



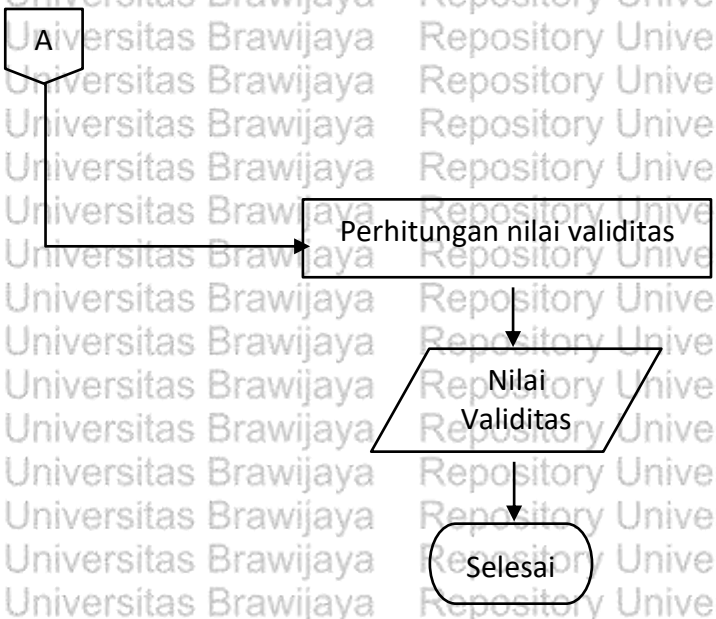
Gambar 4.21 Diagram Alir MKNN

4.6 Hitung Validitas

Pehitungan nilai validitas dilakukan mengacu nilai yang diperoleh dari proses perhitungan BM25. Perhitungan validitas ini bertujuan untuk mengetahui kemiripan kelas pada dokumen data latih dengan tetangganya. Gambar 4.22 menunjukkan proses penghitungan validitas.



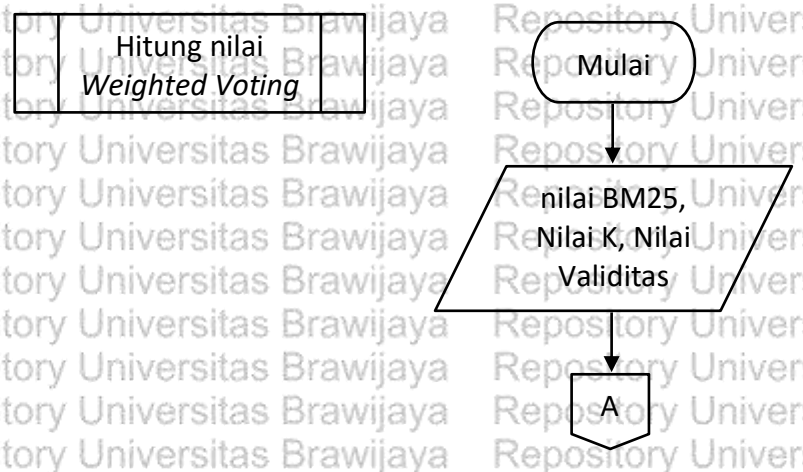
Gambar 4.22 Diagram Alir Hitung Nilai validitas



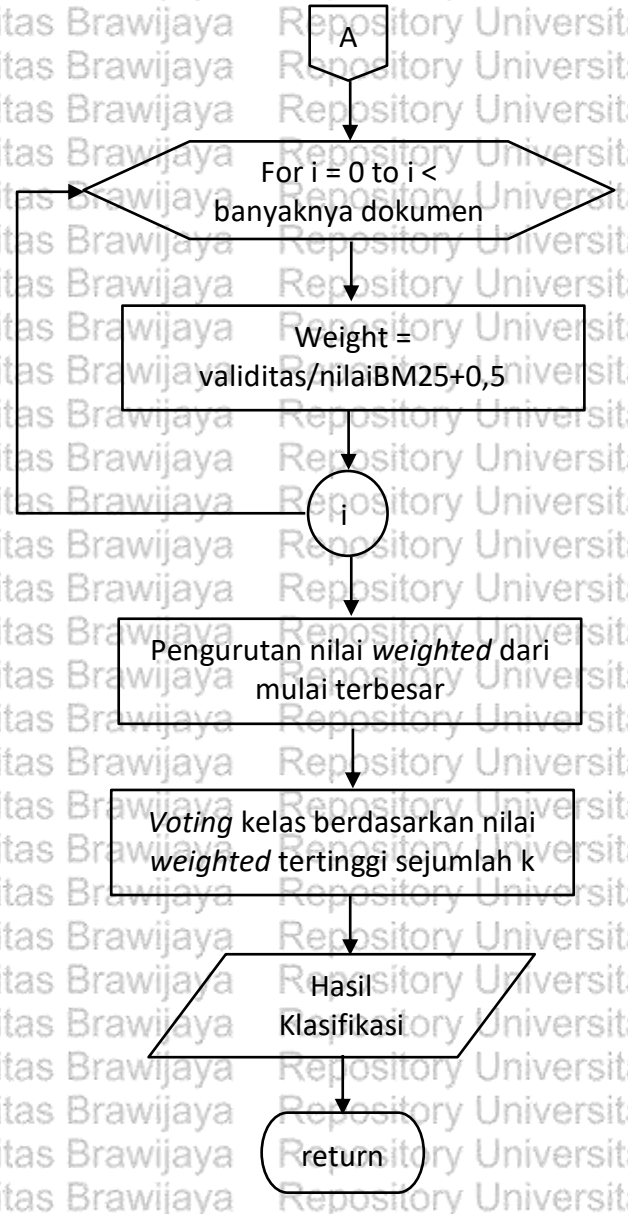
Gambar 4.23 Diagram Alir Hitung Nilai validitas (lanjutan)

4.7 Hitung *Weighted Product*

Perhitungan nilai validitas yang telah didapatkan kemudian menghitung nilai *weight voting* yang bertujuan untuk memperoleh label dari kelas yang ingin diuji. Gambar 4.22 menunjukkan tahapan pada proses *weight voting.*



Gambar 4.24 Diagram Alir *Weighted Voting*



Gambar 4.25 Diagram Alir *Weighted Voting* (lanjutan)

4.8 Manualisasi

Manualisasi adalah proses perhitungan secara manual pada data luji dan data latih. Manualisasi meliputi langkah-langkah *pre-processing,* perhitungan dengan BM25 dan proses klasfikasi dengan MKNN. Sampel data latih dapat dilihat pada Tabel 4.1.

Tabel 4.1 Data Latih

No.	Review	Kelas
1	ini sensasi dingin nya bagus buat wajah yang kemerah-merahan, bisa juga untuk luka bakar	Positif
2	Aku selalu pakai ini sejak muka bruntusan sampai alhamdulillah membaik. Setiap hari aku pake haha. Bagus untuk melembabkan	Positif
3	Semua masalah kulitku teratasi dengan bantuan produk ini, Teksturnya gel jadi enak banget buat kulit oilyku	Positif
4	Pernah pakai ini karna tergiur sama omongan temen, tapi sayang diwajah aku ga cocok malah bikin bruntusan	Negatif
5	aku kurang suka finishnya, kalo pakanya agak banyak, dia malah kering dan ada residunya.	Negatif
6	pakai ini malah jadi beruntusan, jerawatan, muka jadi parah banget	Negatif

Tabel 4.2 Data Uji

No.	Review	Kelas
1	Cinta banget sama produk ini	???

4.8.1 Manualisasi *Case Folding*

Untuk mengubah semua huruf kapital menjadi huruf kecil masuk kedalam proses yang pertama yaitu *case folding*. Proses *case folding* pada data uji dan data latih dapat dilihat di Tabel 4.3 dan 4.4.

Tabel 4.3 *Case Folding* Data Latih

No.	Review	Kelas
1	ini sensasi dingin nya bagus buat wajah yang kemerah-merahan, bisa juga untuk luka bakar	Positif
2	aku selalu pakai ini sejak muka bruntusan sampai alhamdulillah	Positif

	membaik. setiap hari aku pake haha. bagus untuk melembabkan	
3	semua masalah kulitku teratasi dengan bantuan produk ini, teksturnya gel jadi enak banget buat kulit oilyku	Positif
4	pernah pakai ini karna tergiur sama omongan temen, tapi sayang diwajah aku ga cocok malah bikin bruntusan	Negatif
5	aku kurang suka finishnya, kalo pakainya agak banyak, dia malah kering dan ada residunya.	Negatif
6	pakai ini malah jadi beruntusan, jerawatan, muka jadi parah banget	Negatif

Tabel 4.4 *Case Folding* Data Uji

No.	Review	Kelas
1	cinta banget sama produk ini	???

4.8.2 Manualisasi *Cleaning*

Tahapan *cleaning* berguna untuk menghilangkan tanda tanda baca yang tidak dibutuhkan. Proses *cleaning* data latih ditunjukkan di Tabel 4.5 dan Tabel 4.6 untuk data uji.

Tabel 4.5 *Cleaning* Data Latih

No.	Review	Kelas
1	ini sensasi dingin nya bagus buat wajah yang kemerah merahan, bisa juga untuk luka bakar	Positif
2	aku selalu pakai ini sejak muka bruntusan sampai alhamdulillah membaik setiap hari aku pake haha bagus untuk melembabkan	Positif
3	semua masalah kulitku teratasi dengan bantuan produk ini teksturnya gel jadi enak banget buat kulit oilyku	Positif

4	pernah pakai ini karna tergiur sama omongan temen tapi sayang diwajah aku ga cocok malah bikin bruntusan	Negatif
5	aku kurang suka finishnya, kalo pakainya agak banyak dia malah kering dan ada residunya	Negatif
6	pakai ini malah jadi beruntusan jerawatan muka jadi parah banget	Negatif

Tabel 4.6 Cleaning Data Uji

No.	Review	Kelas
1	cinta banget sama produk ini	???

4.8.3 Manualisasi Tokenisasi

Tahapan tokenisasi berguna untuk memisah tiap kalimat atau frase menjadi kata -kata. Proses tokenisasi data latih ditunjukkan pada tabel 4.7 dan tabel 4.8 pada data uji.

Tabel 4.7 Tokenisasi Data Latih

No.	Review	Kelas
1	[‘ini’, ‘sensasi’, ‘dingin’, ‘nya’ , ‘bagus’, ‘buat’, ‘wajah’, ‘yang’, ‘kemerah’, ‘merahan, ‘bisa’, ‘juga’, ‘untuk’, ‘luka’, ‘bakar’]	Positif
2	[‘aku’, ‘selalu’, ‘pakai’, ‘ini’, ‘sejak’, ‘muka’, ‘bruntusan’, ‘sampai’, ‘alhamdulillah’, ‘membaik’, ‘setiap’, ‘hari’, ‘aku’, ‘pake’, ‘haha’, ‘bagus’, ‘untuk’, ‘melembabkan’]	Positif
3	[‘semua’, ‘masalah’, ‘kulitku’, ‘teratasi’, ‘dengan’, ‘bantuan’, ‘produk’, ‘ini’, ‘teksturnya’, ‘gel’, ‘jadi’, ‘enak’, ‘banget’, ‘buat’, ‘kulit’, ‘oilyku’]	Positif
4	[‘pernah’, ‘pakai’, ‘ini’, ‘karna’, ‘tergiur’, ‘sama’, ‘omongan’, ‘temen’, ‘tapi’, ‘sayang’, ‘diwajah’, ‘aku’, ‘ga’, ‘cocok’, ‘malah’, ‘bikin’, ‘bruntusan’]	Negatif
5	[‘aku’, ‘kurang’, ‘suka’, ‘finishnya’, ‘kalo’, ‘pakainya’, ‘kering’, ‘residunya’]	Negatif

6	[‘pakai’, ‘ini’, ‘malah’, ‘jadi’, ‘beruntusan’, ‘jerawatan’, ‘muka’, ‘jadi’, ‘parah’, ‘banget’]	Negatif

Tabel 4.8 Tokenisasi Data Uji

No.	Review	Kelas
1	[‘cinta’, ‘banget’, ‘sama’, ‘produk’, ‘ini’]	???

4.8.4 Manualisasi *Filtering*

Untuk menghapus term yamg tidak dibutuhkan berdasarkan *stoplist,* maka digunakan tahapan *filtering.* Proses *filtering* data latih ditunjukkan pada tabel 4.9 serta tabel 4.10 pada data uji.

Tabel 4.9 *Filtering* Data Latih

No.	Review	Kelas
1	[‘sensasi’, ‘dingin’ , ‘nya’ , ‘bagus’, ‘wajah’, ‘kemerah’, ‘merahan’, ‘luka’, ‘bakar’]	Positif
2	[‘pakai’, ‘muka’, ‘bruntusan’, ‘alhamdulilah, ‘membaik’, ‘pakai’, ‘haha’, ’‘bagus’,‘melembabkan’]	Positif
3	[‘kulitku’, ‘teratasi’, ‘bantuan’, ‘produk’, ‘teksturnya’, ‘gel’, ‘enak’, ‘banget’, ‘kulit’, ‘oilyku’]	Positif
4	[‘pakai’,’karna’, ‘ ‘tergiur’, ‘omongan’, ‘temen’, ‘sayang’, ‘diwajah’,’ga’, ‘cocok’, ‘bikin’, ‘bruntusan’]	Negatif
5	[‘suka’, ‘finishnya’,’kalo’, ‘pakainya’, ‘kering’, ‘residunya’]	Negatif
6	[‘pake’, ‘beruntusan’, ‘jerawatan’, ‘muka’, ‘parah’, ‘banget’]	Negatif

Tabel 4.10 *Filtering* Data Uji

No.	Review	*Stemming*
1	[‘cinta’, ‘banget’, ‘produk’]	???

4.8.5 Manualisasi *Stemming*

Tahapan *stemming* berguna untuk merubah kata berimbuhan menjadi kata dasar. Proses *stemming* data latih dapat dilihat di Tabel 4.11 serta Tabel 4.12 untuk data uji.

Tabel 4.11 *Stemming* Data Latih

No.	Review	Kelas
1	[‘sensasi’, ‘dingin’ , ‘nya’ , ‘bagus’, ‘wajah’, ‘merah’,’merah’, ‘luka’, ‘bakar’]	Positif
2	[‘pakai’, ‘muka’, ‘bruntusan’, ‘alhamdulilah, ‘baik’, ‘pakai’, ‘haha’, ‘’bagus’,’melembabkan’]	Positif
3	[‘kulit’, ‘atas’, ‘bantu’, ‘produk’, ‘tekstur’, ‘gel’, ‘enak’, ‘banget’ , ‘kulit’, ‘oilyku’]	Positif
4	[‘pakai’,’karna’, ‘ ‘giur’, ‘omong’, ‘temen’, ‘sayang’, ‘wajah’,’ga’, ‘cocok’, ‘bikin’, ‘bruntusan’]	Negatif
5	[‘suka’, ‘finishnya’,’kalo’, ‘pakai’, ‘kering’, ‘residu’]	Negatif
6	[‘pakai’, ‘bruntusan’, ‘jerawat’, ‘muka’, ‘parah’, ‘banget’]	Negatif

Tabel 4.12 *Stemming* Data Uji

No.	Review	*Stemming*
1	[‘cinta’, ‘banget’, ‘produk’]	[‘cinta’, ‘banget’, ‘produk’]

4.8.6 Manualisasi BM25

Pada tahap ini dilakukan proses menghitung dengan BM25 yang bertujuan untuk menghitung pemeringkatan. Tahapan pada BM25 adalah menghitung tf, menghitung df menghitung idf, menghitung panjang kalimat, menghitung rata-rata panjang kalimat, menghitung *score* BM25, kemudian dilakukan pemeringkatan.

4.8.6.1 Perhitungan TF

Pada tahap ini memasuki perhitungan dengan BM25 yang memiliki beberapa proses yaitu kemunculan term pada dokumen data uji dihitung,

menghitung kedekatan antara data latih terhadap data uji, dan pemeringkatan terhadap hasil perhitungan data uji dengan data latih. Kemunculan term atau *Term Frequency* dapat dilihat di Tabel 4.13.

Tabel 4.13 Manualisasi *Term Frequency*

Term	D1	D2	D3	D4	D5	D6
sensasi	1	0	0	0	0	0
dingin	1	0	0	0	0	0
bagus	1	1	0	0	0	0
wajah	1	0	0	1	0	0
merah	1	0	0	0	0	0
luka	1	0	0	0	0	0
bakar	1	0	0	0	0	0
pakai	0	2	0	1	1	1
muka	0	1	0	0	0	1
bruntusan	0	1	0	1	0	1
melembabkan	0	1	0	0	0	0
kulit	0	0	1	0	0	0
atas	0	0	1	0	0	0
bantu	0	0	1	0	0	0
produk	0	0	1	0	0	0
tekstur	0	0	1	0	0	0
gel	0	0	1	0	0	0
enak	0	0	1	0	0	0
banget	0	0	1	0	0	1
oilyku	0	0	1	0	0	0
giur	0	0	0	1	0	0
omong	0	0	0	1	0	0
temen	0	0	0	1	0	0
sayang	0	0	0	1	0	0
cocok	0	0	0	1	0	0
suka	0	0	0	0	1	0
finishnya	0	0	0	0	1	0
kering	0	0	0	0	1	0

residu	0	0	0	0	1	0
jerawat	0	0	0	0	0	1
parah	0	0	0	0	0	1
baik	0	1	0	0	0	0
alhamdulilah	0	1	0	0	0	0
bikin	0	0	0	1	0	0
ga	0	0	0	1	0	0
haha	0	1	0	0	0	0
kalo	0	0	0	0	1	0
karna	0	0	0	1	0	0
nya	1	0	0	0	0	0

4.8.6.2 Perhitungan DF

Setelah dilakukan perhitungan *tf* langkah berikutnya yaitu nilai *document frequency* dihitung yang berguna untuk mengetahui berapa *term* yang muncul pada dokumen. Manualisasi *df* dapat dilihat pada Tabel 4.14.

Tabel 4.14 Manualisasi *Document Frequency*

sensasi	1
dingin	1
bagus	2
wajah	2
merah	1
luka	1
bakar	1
pakai	4
muka	2
bruntusan	3
melembabkan	1
kulit	1
atas	1
bantu	1
produk	1
tekstur	1
gel	1

enak	1
banget	2
oilyku	1
giur	1
omong	1
temen	1
sayang	1
cocok	1
suka	1
finishnya	1
kering	1
residu	1
jerawat	1
parah	1
baik	1
alhamdulilah	1
bikin	1
ga	1
haha	1
kalo	1
karna	1
nya	1

4.8.6.3 Perhitungan IDF

Tahapan selanjutnya adalah menghitung *idf.* Perhitungan *idf* meemakai persamaan 2.5. Manualisasi *idf* dapat diliht pada Tabel 4.17. Berikut contoh dari perhitungan *idf* dengan jumlah dokumen sebanyak 6*:*

IDF (sensasi) = $log \frac{N-n(q_1)+0.5}{n(q_1)+0.5}$ = $log \frac{6-1+0.5}{1+0.5}$ = 0,56427143

IDF (dingin) = $log \frac{N-n(q_1)+0.5}{n(q_1)+0.5}$ = $log \frac{6-1+0.5}{1+0.5}$ = 0,56427143

IDF (bagus) = $log \frac{N-n(q_1)+0.5}{n(q_1)+0.5}$ = $log \frac{6-2+0.5}{2+0.5}$ = 0,255272505

Tabel 4.15 Manualisasi IDF

sensasi	0,56427143

dingin	0,56427143
bagus	0,255272505
wajah	0,255272505
merah	0,56427143
luka	0,56427143
bakar	0,56427143
pakai	-0,255272505
muka	0,255272505
bruntusan	0
melembabkan	0,56427143
kulit	0,56427143
atas	0,56427143
bantu	0,56427143
produk	0,56427143
tekstur	0,56427143
gel	0,56427143
enak	0,56427143
banget	0,255272505
oilyku	0,56427143
giur	0,56427143
omong	0,56427143
temen	0,56427143
sayang	0,56427143
cocok	0,56427143
suka	0,56427143
finishnya	0,56427143
kering	0,56427143
residu	0,56427143
jerawat	0,56427143
parah	0,56427143
baik	0,56427143
alhamdulilah	0,56427143
bikin	0,56427143

ga	0,56427143
haha	0,56427143
kalo	0,56427143
karna	0,56427143
nya	0,56427143

4.8.6.4 Perhitungan Panjang Kalimat

Langkah berikutnya yaitu menghitung panjang kalimat dari setiap kata yang terdapat pada data latih. Hasil manualisasi dari panjang dilihat pada Tabel 4.16.

Tabel 4.16 Manualisasi Panjang Kalimat

	D1	D2	D3	D4	D5	D6
Panjang Data	9	9	10	11	6	6

4.8.6.5 Perhitungan Rata-Rata Panjang Kalimat

Selanjutnya setelah didapat nilai dari tiap panjang kalimat, dihitung rata-ratanya. Berikut merupakan contoh perhitungan rata-rata dari panjang kalimat yang telah didapat:

$$Avg = \frac{9+9+10+11+6+6}{6} = 8{,}5$$

4.8.6.6 Perhitungan *Score* BM25

Langkah selanjutnya adalah menghitung nilai bobot dengan BM25, pada tahap ini dibutuhkan nilai k dan b yaitu sebesar 1,5 dan 0,75. Untuk menghitung *score* BM25 digunakan Persamaan 2.5. Berikut contoh perhitungan *score* BM25.

$$\text{Wd (sensasi,D1)} = \frac{tf(q_i,d).(k_1+1)}{tf(q_i,d)+k_i.(1-b+b.\frac{|d|}{dl_{avg}})}$$

$$= \frac{1.(1{,}5+1)}{1+1{,}5.(1-0{,}75+0{,}75.\frac{9}{8{,}5})} = 0{,}97421$$

$$\text{Wd (dingin,D1)} = idf(q_i).\frac{tf(q_i,d).(k_1+1)}{tf(q_i,d)+k_i.(1-b+b.\frac{|d|}{dl_{avg}})}$$

$$= \frac{1.(1{,}5+1)}{1+1{,}5.(1-0{,}75+0{,}75.\frac{9}{8{,}5})} = 0{,}97421$$

$$\text{Wd (bagus,D1)} = \frac{tf(q_i,d).(k_1+1)}{tf(q_i,d)+k_i.(1-b+b.\frac{|d|}{dl_{avg}})}$$

$$= \frac{1.(1{,}5+1)}{1+1{,}5.(1-0{,}75+0{,}75.\frac{9}{8{,}5})} = 0{,}97421$$

Tabel 4.17 Manualisasi nilai Wd

	Wd					
Term	Wd1	Wd2	Wd3	Wd4	Wd5	Wd6
sensasi	0,9742	0	0	0	0	0
dingin	0,9742	0	0	0	0	0
bagus	0,9742	0,9742	0	0	0	0
wajah	0,9742	0	0	0,8831168	0	0
merah	0,9742	0	0	0	0	0
luka	0,9742	0	0	0	0	0
bakar	0,9742	0	0	0	0	0
pakai	0	1,4020	0	0,8831168	1,1525	1,1525
muka	0	0,9742	0	0	0	1,1525
bruntusan	0	0,9742	0	0,8831168	0	1,1525
melembabkan	0	0,9742	0	0	0	0
kulit	0	0	0,926430	0	0	0
atas	0	0	0,926430	0	0	0
bantu	0	0	0,926430	0	0	0
produk	0	0	0,926430	0	0	0
tekstur	0	0	0,926430	0	0	0
gel	0	0	0,926430	0	0	0
enak	0	0	0,926430	0	0	0
banget	0	0	0,926430	0	0	1,1525
oilyku	0	0	0,92643052	0	0	0
giur	0	0	0	0,8831168	0	0
omong	0	0	0	0,8831168	0	0
temen	0	0	0	0,8831168	0	0
sayang	0	0	0	0,8831168	0	0

cocok	0	0	0	0,883116883	0	0
suka	0	0	0	0	1,1525	0
finishnya	0	0	0	0	1,1525	0
kering	0	0	0	0	1,1525	0
residu	0	0	0	0	1,1525	0
jerawat	0	0	0	0	0	1,1525
parah	0	0	0	0	0	1,1525
baik	0		0	0	0	0
alhamdulilah	0	0,9742	0	0	0	0
bikin	0		0	0,88311688	0	0
ga	0		0	0,88311688	0	0
haha	0	0,9742	0	0	0	0
kalo	0	0	0	0	1,1525	0
karna	0	0	0	0,88311688	0	0
nya	0,974212	0	0	0	0	0

Setelah didapat nilai *wd* pada Tabel 4.17 selanjutnya dikalikan dengan *idf* yang telah dihitung sebelumnya. Sehingga menghasilkan nilai BM25 untuk setiap data latih yang ada di Tabel 4.18. Contoh perhitungan antar data latih:

$$BM25(D2, D1) = \ Wd_{D3}(bagus) \times idf(bagus)$$

$$BM25(D2, D1) = 0{,}97421 \times 0{,}255275 = 0{,}2486895$$

Tabel 4.18 Manualisasi BM25 Antar Data Latih

BM25	D1	D2	D3	D4	D5	D6
D1		0,2486895	0	0,22543546	0	0
D2	0,24868955		0	-0,2254355	-0,2942124	0
D3	0	0		0	0	0,29421237
D4	0,2486895	-0,357907	0		-0,294212	-0,2942123

D5	0	-0,357907	0	-0,2254355		-0,2942123
D6	0	-0,109218	0,23649	-0,2254355	-0,294212	

4.8.6.7 Manualisasi BM25 Antar Data Uji

Sama hal nya sebagaimana di data latih, pada data uji tahapan perhitungannya yang pertama menghitung *tf*,*df*,dan *idf* kemudian dihitung nilai *wd* nya lalu nilai wd yang didapatkan dikalikan dengan nilai *idf*. Kemudian nilai BM25 pada setiap term dijumlahkan dan menjadi nilai BM25 dari dokumen data uji tersebut. Manualisasi untuk perhitungan BM25 pada data uji bisadilihat pada Tabel 4.19.

$$BM25(D3,Q) = Wd_{D3}(banget) \times idf(banget) + Wd_{D3}(produk) \times idf(produk) = 0{,}926431 \times 0{,}25527 + 0{,}926431 \times 0{,}56427 = 0{,}75925$$

Tabel 4.19 Manualisasi BM25 Data Uji

BM25 DATA UJI						
	D1	D2	D3	D4	D5	D6
Q	0	0	0,75925	0	0	0,294212379

4.8.7 Manualisasi MKNN

Setelah menghitung pembobotan dengan BM25 selanjutnya adalah mengklasifikasikan kelas dengan MKNN.

4.8.7.1 Manualisasi Nilai Validitas

Setelah dilakukan pembobotan dengan BM25 selanjutnya dilakukan perhitungan nilai validitas terhadap data sejumlah k yang merupakan data dengan nilai BM25 paling tinggi terhadap data tersebut. Untuk mengetahui hasil validitas digunakan Persamaan 2.2. Dokumen terdekat sejumlah k ditunjukkan di tabel 4.20.

Tabel 4.20 Tetangga Terdekat Sejumlah K Pada Data Latih

Dokumen	K=3		
D1	D2	D4	D3
D2	D1	D3	D5
D3	D6	D1	D5
D4	D1	D3	D5
D5	D1	D3	D4
D6	D3	D1	D2

Kemudian dihitung nilai validitasnya

$$Validitas\ (D1) = \frac{1}{3} \times (1 + 0 + 1) = 0{,}666667$$

Perhitungan nilai validitas ditunjukkan pada tabel 4.23

Tabel 4.21 Manualisasi Nilai Validitas

Dokumen	k=3			Validitas
D1	1	0	1	0,666667
D2	1	1	0	0,666667
D3	0	1	0	0,333333
D4	1	1	0	0,333333
D5	1	1	0	0.333333
D6	1	1	1	0

4.8.7.2 Manualisasi Nilai *Weight Voting*

Perhitungan proses *weight voting* bertujuan menetukan kelas pada suatu data uji dengan cara melakukan *voting* pada data sejumlah k. pada proses ini pertama dilakukan perhitungan nilai bobot pada data uji. Kemudian dari nilai bobot tersebut diurutkan dari nilai yang paling terbesar. Lalu dilakukan *vote* dengan cara menjumlahkan nilai bobot apabila terdapat kelas yang sama dengan data yang ada. *Vote* tertinggi merupakan kelas hasil perhitungan proses *weight voting*. Manualisasi nilai *voting* ditunjukkan di tabel 4.22.

$$W(D1) = Validity(D1) \times \frac{1}{BM25_{(D1,Q)} + 0.5}$$

$$W(D1) = 0{,}666667 \times \frac{1}{0 + 0.5} = 1{,}33333$$

Tabel 4.22 Manualisasi *Weight Voting*

WEIGHT VOTING	
Dokumen	Q
D1	1,333333333
D2	1,333333333
D3	0,264707721
D4	0,666666667
D5	0,666666667

D6	0

Setelah diurutkan ditunjukkan pada tabel 4.23

Tabel 4.23 Pengurutan Manualisasi *Weight Voting*

WEIGHT VOTING		
Dokumen	kelas	Q
D1	Positif	1,333333333
D2	Positif	1,333333333
D4	Negatif	0,666666667
D5	Negatif	0,666666667
D3	Negatif	0,264707721
D6	Negatif	0

$W(Positif) = 1{,}333333 + 1{,}333333 = 2{,}666667$

$W(Negatif) = 0{,}6666667$

Karena nilai $W(Positif) > W(Negatif)$, maka kelas data uji adalah "Positif"

4.9 Perancangan Pengujian

Perancangan pengujian ini dirancang untuk menentukan hasil evaluasi sistem. Pada tahap ini dijelaskan mengenai pengujian apa saja yang dilakukan.

4.9.1 Perancangan Pengaruh K-*Fold* dan Nilai K

Perancangan menggunakan *k-fold* serta nilai *k* tujuannya yaitu untuk menentukan akurasi, akankah nilai k mempengaruhi hasil sistem klasifikasi. Perancangan pengaruh K-*fold* dan nilai *k* ditunjukkan di tabel 4.24

Tabel 4.24 Perancangan Pengaruh *K-Fold* dan *k*

Nilai K	Hasil Precision (%)	Hasil Recall (%)	Hasil F-Measure	Hasil Akurasi (%)
3				
5				
7				
9				

11				
13				
15				
27				
57				
107				

BAB 5 IMPLEMENTASI

Didalam bab Implementasi ini menjelaskan bagaimana implementasi dari kode program yang telah dibuat. Dalam pengimplementasian sistem ini digunakan Bahasa pemrograman *python* pada proses pelatihan dan pengujian.

5.1 Lingkungan Pengembangan Sistem

Lingkungan pengembanggan pada sistem digunakan untuk memahami perangkat lunak dan perangkat keras yang diperlukan untuk membangun sistem ini dan menyusun kode program sehingga dapat diimplementasikan pada sistem ini.

a) Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan penulis disajikan pada Tabel 5.1

Tabel 5.1 Perangkat Keras

No	Keterangan	Detil
1	*Processor*	Intel core i3
2	*Memory*	4Gb

b) Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan penulis disajikan pada Tabel 5.2

Tabel 5.2 Perangkat Lunak

No	Keterangan	Detil
1	Sistem Operasi	Windows 10
2	IDE	Spyder (Python 3.7)

5.2 Implementasi Algoritma

Implementasi algoritma memaparkan mengenai tahapan yang mesti dilakukan pada Sistem Klasifikasi *Review* Produk Kecantikan Pada Aplikasi *Sociolla* menggunakan metode MKNN dan Pembobotan BM25. Didalam proses implementasi dilakukkan berdasar pada proses perancangan yang telah dibahas sebelumnya.

5.2.1 Implementasi *Preprocessing*

Preprocessing merupakan awal dari serangkaian proses yang bertujuan memperoleh term pada proses lanjutan. Tahapan-tahapan pada *preprocessing* meliputi *Cleansing, Case Folding,* Tokenisassi, *Filtering, Stemming*.

1. Proses *Data Cleansing*

Cleansing merupakan sub proses pada tahapan *Preprocessing.* Tahapan ini bertujuan untuk menghilangkan tanda baca, serta simbol-simbol yang tidak diperlukan. Implementasi data cleansing dapat dilihat di Tabel 5.3.

Tabel 5.3 Implmentasi Data Cleansing

No.	Kode Program
1	def dataCleansing(self, data):
2	for i in range(len(data)):
3	data[i] = data[i].replace("-", " ")
4	data[i] = data[i].replace(",", " ")
5	table = str.maketrans(dict.fromkeys(string.punctuation))
6	data[i] = data[i].translate(table)
7	data[i] = re.sub(r"(^|\W)\d+", "", data[i])
8	
9	return data
10	

Method *dataCleansing* menerima parameter berupa list yang berisi data dari kolom 'text' dari file data latih. Perulangan for disini digunakan untuk melakukan looping sebanyak panjang list tersebut. Perulangan tersebut menghilangan tanda baca sepeerti '-' , ',' , dll serta menghilangkan angka.

2. Proses *Case Folding*

Tahapan selanjutnya adalah melakukan *Case Folding* yang bertujuan untuk merubah semua huruf capital menjadi huruff kecil. Implementasi *case folding* dapat dilihat di Tabel 5.4

Tabel 5.4 Implementasi *Case Folding*

No.	Kode Program
1	def caseFolding(self, data):
2	return [x.lower() for x in data]

Method case folding menerima parameter dari hasil method data cleansing. Method ini mengembalikan data yang sama akan tetapi dalam bentuk lower case.

3. Proses Tokenisasi

Pada tahap tokenisasi tujuannya untuk memotong tiap kalimat menjadi kata-kata. Implementasi data tokenisasi dapat dilihat di Tabel 5.5.

Tabel 5.5 Implementasi Tokenisasi

No.	Kode Program
1	def tokenizing(self, data):
2	hasil = []
3	for i in data:
4	hasil.append(i.split(" "))
5	
6	return hasil

Method tokenizing menerima parameter dari hasil method case folding. Method ini terdapat perulangan yang berguna untuk memisahkan kata dalam data yang diterima.

4. Proses *Filtering*

Pada tahap ini yaitu proses menghapus berbagai kata yang ada pada *stoplist.* Implementasi *filtering* dapat dilihat di Tabel 5.6.

Tabel 5.6 Implementasi *Filtering*

No.	Kode Program
1	def removeStopword(self, data):
2	stopword = pd.read_csv('Dataset/tala.csv')
3	
4	hasil = []
5	for i in range(len(data)):
6	hasil.append([])
7	for j in range(len(data[i])):
8	if data[i][j] not in stopword.values and data[i][j] != '':
9	hasil[i].append(data[i][j])
10	
11	return hasil
12	

Method removeStopword menerima parameter dari hasil method tokenizing. Method ini membaca file csv stopword tala, kemudian terdapat perulangan sebanyak panjang data parameter yang didalamnya akan dilakukan pemeriksaan adanya kata yang ada didalam file csv stopword tala tersebut, apabila ada maka kata tersebut akan ditambahkan ke sebuah list kosong bernama hasil yang menampung term setelah filtering.

5. Proses *Stemming*

Tahap selanjutnya adalah proses *stemming,* yaitu menghilangkan kalimat yang memiliki imbuhan ke kata dasar. Implementasi *stemming* bisa dilihat di Tabel 5.7

Tabel 5.7 Implementasi *Stemming*

No.	Kode Program
1	def stemming(self, data):
2	factory = StemmerFactory()
3	stemmer = factory.create_stemmer()
4	
5	for i in range(len(data)):
6	for j in range(len(data[i])):
7	data[i][j] = stemmer.stem(data[i][j])
8	
9	return data

Method stemming menerima parameter berupa data dari hasil *method* removeStopword. Method ini memanggil *stemmer factory* dari *library stemmer* sastrawi untung melakukan *stemming*. Selanjutnya dilakukan perulangan for sebanyak panjang list yang diterima yang digunakan untuk melakukan stemming pada tiap term yang ada dalam list tersebut.

6. Proses mendapatkan term akhir

Setelah dilakukan tahapan-tahapan pada *preprocessing* , untuk mendapatkan term akhir digunakan method getTerm seperti pada Tabel 5.8.

Tabel 5.8 Implementasi Mendapatkan *Term* Akhir

No.	Kode Program
1	def getTerm(self, data):
2	hasil = []
3	for i in range(len(data)):
4	for j in range(len(data[i])):
5	hasil.append(data[i][j])
6	
7	return np.unique(hasil)

Method getTerm menerima parameter dari data hasil method stemming. Pada method ini dilakukan perulangan sebanyak panjang list yang diterima.

Method ini berguna untuk menghapus duplikasi dengan cara menggunakan method unique yg ada pada library numpy.

5.2.2 Implementasi Perhitungan Bobot

Setelah dilakukan *PreProcessing,* dilakukan perhitungan bobot. Implementasi untuk mendapatkan *rawTF* dapat dilihat pada Tabel 5.9.

1. Proses perhitungan nilai *Term Frequency*

Tabel 5.9 Implementasi *Raw Term Frequency*

No.	Kode Program
1	def getRawTF(self, data, term):
2	hasil = np.zeros((len(term), len(data)))
3	
4	for i in range(len(term)):
5	for j in range(len(data)):
6	for k in range(len(data[j])):
7	if term[i] == data[j][k]:
8	hasil[i][j] +=1
9	
10	return hasil

Method getRawTF menerima parameter berupa data dari hasil *preprocessing.* Pada method ini dilakukan perulangan sebanyak panjang list yang diterima. Method ini berguna untuk mendapatkan hasil *raw TF.*

2. Proses perhitungan nilai *Document Frequency*

Setelah didapatkan rarwTF selanjutnya masuk pada proses *Document Frequency.* Implementasi *document frequency* dapat dilihat pada Tabel 5.10.

Tabel 5.10 Implementasi *Document Frequency*

No.	Kode Program
1	def getDocumentFrecuency(self, doc):
2	hasil = doc.copy()
3	for i in range(len(hasil)):
4	for j in range(len(hasil[i])):
5	if hasil[i][j] >= 1:
6	hasil[i][j] = 1
7	return hasil

Method getDocumentFrecuency menerima parameter berupa data dari hasil rawTF. Didalam method ini dilakukan perulangan sebanyak panjang *list* yang diterima. Method ini berguna untuk mendapat kan hasil *df* atau *document frequency.*

3. Proses perhitungan nilai *Inverse Document Frequency*

Setelah didapatkan nilai df maka selanjutnya dihitung nilai idf yang bisa dilihat di Tabel 5.11. Implementasi *document frequency* bisa dilihat di Tabel 5.10.

Tabel 5.11 Implementasi *Inverse Document Frequency*

No.	Kode Program
1	def getIdf(self, docs):
2	idf = np.zeros(len(docs))
3	df = np.zeros(len(docs))
4	
5	for i in range(len(docs)):
6	df[i] = np.sum(docs[i])
7	
8	if (df[i] != 0):
9	idf[i] = np.log10((len(docs[i])-df[i]+0.5)/(df[i]+0.5))
10	else:
11	idf[i] = 0
12	
13	return df, idf
14	

Method getIdf menerima parameter dari data hasil *df* pada method ini dilakukan perhitungan menggunakan rumus pada Persamaan 2.6. Method ini berguna untuk mendapatkan hasil *idf.*

4. Proses perhitungan nilai rata-rata dari panjang dokumen.

Langkah selanjutnya dilakukan perhitungan IDF langkah selanjutnya adalah menghitung panjang kalimat yaitu menghitung seluruh total kata didalam dokumen. Implementasi menghitung panjang dokumen bisa dilihat di Tabel 5.12.

Tabel 5.12 Implementasi Menghitung Panjang Dokumen

No.	Kode Program
1	def getLengthOfDocuments(self,data):
2	length_of_documents = np.zeros(len(data))
3	

```
4       for i in range(len(data)):
5           length_of_documents[i] = len(data[i])
6       print ("Panjang Dokumen: ")
7       print (length_of_documents)
8
9       return length_of_documents, np.average(length_of_documents)
10
```

Method getLengthOfDocuments berguna untuk menghitung panjang dokumen yang ada. Awalnya dilakukan perulangan untuk menghitung panjang dokumennya kemudian masuk ke perhitungan agar rata-rata dari panjang dokumen bisa dihitung.

5.2.3 Implementasi Perhitungan Nilai *Score* BM25

1. Proses Perhitungan nilai *Wd*

Selanjutnya adalah menghitung nilai bobot dengan BM25 atau bisa disebut nilai *Wd*. Implementasi nilai Wd bisa dilihat di Tabel 5.13.

Tabel 5.13 Implementasi Hitung *Wd*

No.	Kode Program
1	def getWd(self, raw_tf, length_of_documents, average_of_documents, k, b):
2	wd = np.zeros((len(raw_tf), len(raw_tf[0])))
3	
4	for i in range(len(wd)):
5	for j in range(len(wd[i])):
6	wd[i][j] = (raw_tf[i][j] * (k+1)) / (raw_tf[i][j] + k*((1-b) +
7	b*length_of_documents[j]/average_of_documents))
8	
9	return wd
10	

Method getWd menerima parameter berupa *tf*, panjang dok, rata-rata dokumen, variabel *k* dan *b*. Pertama-tama membuat list sebesar *list raw_tf,* kemudian dilakukan perulangan *nested for* untuk menghitung *wd* di tiap data *tf*. Lalu sesuai dengan rumus yang ada di Persamaan 2.5 dilakukan perhitungan.

2. Perhitungan Score BM25

Setelah didapat nilai *wd,* dikalikan dengan *idf* yang telah dihitung sebelumnya. Impementasinya bisa dilihat di Tabel 5.14.

Tabel 5.14 Implementasi Hitung *Score* BM25

No.	Kode Program
1	def hitungBM25(self,data_sim, terms, idf, wd):
2	result = 0
3	for i in range(len(data_sim)):
4	for j in range(len(terms)):
5	if data_sim[i] == terms[j]: result += idf[j] * wd[j]
6	return result
7	

Setelah didapatkan nilai *wd* dan *idf* yang telah dihitung maka masuk pada *method* hitungBM25. Pada *method* ini menerima parameter berupa list berisi *similarity check*, *term-term* setelah *preprocessing*, nilai idf dalam *list idf*, dan *nilai wd* dalam *list wd*, method ini mengembalikan nilai variable result. Didalam *method* ini terdapat perulangan *nested for* sebanyak panjang list data_sim dan jumlah tem dalam *list terms*. Jika data ke-i di data_sim sama dengan data ke-j di *list terms* maka result ditambah dg nilai *idf* ke-*j* dan *wd* ke-j.

3. Perhitungan Nilai BM25 antar data uji serta data latih

Langkah selanjutnya akan menghasilkan nilai BM25 antar data latih. Implementasinya bisa dilihat di Tabel 5.15.

Tabel 5.15 Impementasi Score BM25 Antar Data Uji dan Data Latih

No.	Kode Program
1	def getBM25(self, params, idf, wd):
2	bm25_latih= np.zeros((len(params['data_latih_pre']),len(params['data_latih_pre'])))
3	bm25_uji= np.zeros((len(params['data_uji_pre']),len(params['data_latih_pre'])))
4	similarity_chek_uji = []
5	for i in range(len(params['data_uji_pre'])):
6	similarity_chek_uji.append([])
7	for j in range(len(params['data_latih_pre'])):
8	sim = self.getSimilarityCheck(params['data_uji_pre'][i], params['data_latih_pre'][j])
9	similarity_chek_uji[i].append(sim)
10	similarity_chek_latih = []
11	for i in range(len(params['data_latih_pre'])):
12	similarity_chek_latih.append([])
13	for j in range(len(params['data_latih_pre'])):
14	if (i != j):
	sim = self.getSimilarityCheck(params['data_latih_pre'][i], params['data_latih_pre'][j])
	sim = np.unique(sim)

```
                similarity_chek_latih[i].append(sim)
            else:
                similarity_chek_latih[i].append([])

for i in range(len(bm25_uji)):
    for j in range(len(bm25_uji[i])):
        bm25_uji[i][j] = self.hitungBM25(similarity_chek_uji[i][j], params['term_latih'], idf, wd[:,j])

for i in range(len(bm25_latih)):
    for j in range(len(bm25_latih[i])):
        bm25_latih[i][j] = self.hitungBM25(similarity_chek_latih[i][j], params['term_latih'], idf, wd[:,j])

return bm25_latih, bm25_uji
```

Method ini menerima parameter berupa hasil prepocesing data latih, hasi preprocessing data uji, dimana keduanya sudah berupa list per dokumen yang terdiri atas sublist berisi term per dokumen, serta list IDF dan Wd. Selanjutnya akan ada perulangan nested for yang digunakan untuk mengecek keberadaan term yang sama dari doc uji dan doc latih, dengan memanggil method getSimilarityCheck. hasil kesamaannya akan ditambahkan ke list similarity_chek_uji dan similarity_check_latih. Dari hasil list similarity_chek_uji dan similarity_chek_latih, maka dapat dihitung BM25nya, dengan memanggil method hitungBM25. Hasil dari perhitungan bm25 tiap term akan dimasukkan ke list bm25_latih untuk term latih dan bm25_uji untuk term uji. Method ini mengembalikan list bm25_latih dan bm25_uji.

5.2.4 Implementasi Perhitungan MKNN

1. Perhitungan nilai Validitas

Setelah dilakukan pembobotan dengan BM25 selanjutnya dilakukan perhitungan nilai validitas terhadap data sejumlah k yang merupakan data dengan nilai BM25 paling tinggi terhadap data tersebut. Implementasinya bisa dilihat di Tabel 5.16.

Tabel 5.16 Implementasi Hitung Validitas

No	Kode Program

```
1  def hitungValiditas(self, k, x_train, y_train):
2      result = []
3      print()
4      for i in range(len(x_train)):
5          data_terhapus, kelas_terhapus  = self.hapusData(i, x_train[i], y_train)
6          data_terurut, kelas_terurut = self.urutkanData(i, data_terhapus,
7  kelas_terhapus)
8          data_terurut  = data_terurut[:k]
9          kelas_terurut  = kelas_terurut[:k]
10         tot = 0
11         for j in range(len(data_terurut)):
12             if (y_train[i] == kelas_terurut[j]):
13                 tot += 1
14         result.append(tot/k)
15     return result
```

Metode ini berguna untuk menghitung nilai validitas pada Algoritme MKNN. Metode ini menerima parameter berupa *data Training* . Dengan persamaan 2.2 untuk memperoleh hasil validitas.

2. Perhitungan nilai *Weight Voting*

Dengan perhitungan menggunakan *weight* voting yang mana di tujukan agar dapat menetukan suatu klasifikasi pada uji data dengan cara melakukan *voting* pada data sejumlah k. Implementasinya bisa dilihat di Tabel 5.17.

Tabel 5.17 Implementasi Hitung *Weight Voting*

No	Kode Program

```
1  def hitungWeightVoting(self, x_test, validitas):
2      result = np.zeros((len(x_test), len(x_test[0])))
3      for i in range(len(x_test)):
4          for j in range(len(x_test[i])):
5              result[i][j] = validitas[j] * (1/(x_test[i][j]+0.5))
6      return result
7
```

Method ini dipakai guna menghitung *weight voting* tiap dokumen data latih sesuai rumus *weight voting* berdasarkan nilai validitasnya. Method ini menerima parameter dari data uji dan nilai validitas yang telah didapatkan. Untuk menghitung nilai weight voting memakai rumus yang ada di Persamaan 2.4.

BAB 6 PENGUJIAN

BAB 6 berisi tentang hasil dari pengujian yang sudah dirancang sebelumnya dan juga dibahas mengenai analisis terhadap hasil yang diperoleh dalam pengujian ini.

6.1 Pengujian Sistem

Sub bab ini menjelaskan mengenai hasil pengujian sistem yang sudah dilakukan. Nilai k dengan 5 *fold cross validation* yang diuji dengan pengujian sistem dengan total data sebanyak 500 yang nantinya akan menghasilkan *f-measure, precision*, nilai akurasi dan *recall.* Pengujian tersebut akan dilakukan sebanyak 5 kali dengan 5 *fold* data yang berbeda-beda. Setelah itu akan dilihat nilai akurasi manakah yang paling tinggi pada nilai k.

6.2 Pengujian Nilai K dan 5-*Fold Cross Validation*

Pengujian nilai k membagi data menjadi 5 *fold* yang mana tiap *fold* datanya digunakan untuk data uji, sedangkan sisa dari *fold* data digunakan sebagai data latih. Nilai k menguji masing-masing *fold* data yang ada dan mencari *precision, f-measure,* nilai akurasi, dan *recall.*

6.2.1 *Fold* ke-1

Data ke 1 hingga 100 pada *fold* data ini digunakan untuk data uji dan sisa data tersebut digunakan sebagai data latih. Pengujian *fold* ke-1 memiliki hasil yang dapat dilihat pada tabel 6.1.

Tabel 6.1 Pengujian *Fold* Ke-1

Nilai K	Hasil Precision (%)	Hasil Recall (%)	Hasil F-Measure (%)	Hasil Akurasi (%)
3	42,22	38,00	40,00	43,00
5	42,22	38,00	40,00	43,00
7	44,44	40,00	42,10	45,00
9	45,65	42,00	43,75	46,00
11	48,97	48,00	48,48	49,00
13	45,65	42,00	43,75	46,00
15	44,44	40,00	42,10	45,00
27	42,85	36,00	39,13	44,00
57	43,90	36,00	39,56	45,00
107	46,51	40,00	43,01	47,00

Nilai k=11 menghasilkan akurasi,presisi,*recall*, serta *f-measure* yang paling tinggi, yaitu 49,00%, 48,97%, 48,00%, dan 48,48% yang dijelaskan pada tabel 6.1.

6.2.2 *Fold* ke-2

Data ke 101 hingga 200 pada *fold* data ini digunakan untuk data uji dan sisa data tersebut digunakan sebagai data latih. Pengujian *fold* ke-2 memiliki hasil yang dapat dilihat pada tabel 6.2.

Tabel 6.2 Pengujian *Fold* Ke-2

Nilai K	Hasil Precision (%)	Hasil Recall (%)	Hasil F-Measure	Hasil Akurasi (%)
3	46,67	42,00	44,21	47,00
5	43,75	42,00	42,85	44,00
7	46,93	46,00	46,46	47,00
9	46,93	46,00	46,00	47,00
11	46,00	46,00	46,46	46,00
13	44,89	44,00	44,44	45,00
15	44,00	44,00	44,00	44,00
27	42,30	44,00	43,13	42,00
57	43,75	42,00	42,85	44,00
107	42,85	42,00	42,42	43,00

Nilai k=7 menghasilkan akurasi,presisi,*recall*, serta *f-measure* yang paling tinggi, yaitu 47,00%, 46,93%, 46,00%, dan 46,46% yang dijelaskan pada tabel 6.2.

6.2.3 *Fold* ke-3

Data ke 201 hingga 300 pada *fold* data ini digunakan untuk data uji dan sisa data tersebut digunakan sebagai data latih. Pengujian *fold* ke-3 memiliki hasil yang dapat dilihat pada tabel 6.3.

Tabel 6.3 Pengujian *Fold* Ke-3

Nilai K	Hasil Precision (%)	Hasil Recall (%)	Hasil F-Measure	Hasil Akurasi (%)
3	55,55	60,00	57,69	56,00

5	52,83	56,00	54,36	53,00
7	50,98	52,00	51,48	51,00
9	52,72	58,00	55,23	53,00
11	52,63	60,00	56,07	53,00
13	52,83	56,00	54,36	53,00
15	51,85	56,00	53,84	52,00
27	51,85	56,00	53,84	52,00
57	51,85	56,00	53,84	52,00
107	50,94	54,00	52,42	51,00

Nilai k=11 menghasilkan akurasi,presisi,*recall*, serta *f-measure* yang paling tinggi, yaitu 53,00%, 52,63%, 60,00%, dan 56,07% yang dijelaskan pada tabel 6.3.

6.2.4 *Fold* ke-4

Data ke 301 hingga 400 pada *fold* data ini digunakan untuk data uji dan sisa data tersebut digunakan sebagai data latih. Pengujian *fold* ke-4 memiliki hasil yang dapat dilihat pada tabel 6.4.

Tabel 6.4 Pengujian *Fold* Ke-4

Nilai K	Hasil Precision (%)	Hasil Recall (%)	Hasil F-Measure	Hasil Akurasi (%)
3	52,00	53,06	52,52	52,00
5	52,08	51,02	51,54	52,00
7	54,16	53,06	53,60	54,00
9	52,00	53,06	52,52	52,00
11	53,06	53,06	53,06	53,00
13	48,97	48,97	48,97	49,00
15	47,91	46,93	47,42	48,00
27	46,80	44,89	45,83	47,00
57	47,91	46,93	47,42	48,00
107	48,00	48,97	48,48	48,00

Nilai k=7 menghasilkan akurasi,presisi,*recall*, serta *f-measure* yang paling tinggi, yaitu 52,00%, 54,16%, 53,06%, dan 53,60% yang dijelaskan pada tabel 6.4.

6.2.5 *Fold* ke-5

Data ke 401 hingga 500 pada *fold* data ini digunakan untuk data uji dan sisa data tersebut digunakan sebagai data latih. Pengujian *fold* ke-5 memiliki hasil yang dapat dilihat pada tabel 6.5.

Tabel 6.5 Pengujian *Fold* Ke-5

Nilai K	Hasil Precision (%)	Hasil Recall (%)	Hasil F-Measure	Hasil Akurasi (%)
3	52,94	54,00	53,46	53,00
5	53,70	58,00	55,76	54,00
7	53,84	56,00	54,90	54,00
9	52,83	56,00	54,36	53,00
11	53,84	56,00	54,90	54,00
13	52,94	54,00	53,46	53,00
15	53,84	56,00	54,90	54,00
27	52,83	56,00	54,36	53,00
57	56,25	54,00	55,10	56,00
107	55,10	54,00	54,54	55,00

Nilai k=7 & k=11 akurasi,presisi,*recall*, serta *f-measure* yang paling tinggi, yaitu 54,00%, 53,84%, 56,00%, dan 54,90% yang dijelaskan pada tabel 6.5.

6.2.6 Rata-rata Hasil Pengujian 5 *Fold*

Setelah nilai k diuji pada 5 *fold* data, kemudian dihitung rata-rata dari keseluruhan hasil yang didapatkan pada tabel 6.6 .

Tabel 6.6 Rata-Rata Pengujian 5-*Fold*

Nilai K	Hasil Precision (%)	Hasil Recall (%)	Hasil F-Measure	Hasil Akurasi (%)
3	49,87	49,41	49,57	50,20
5	48,91	49,00	48,90	49,20
7	50,07	49,41	49,71	50,20
9	50,02	51,01	50,46	50,20
11	50,90	52,61	51,70	51,00

13	49,06	48,99	49,00	49,20
15	48,41	48,58	48,45	48,60
27	47,33	47,37	47,26	47,60
57	48,73	46,98	47,75	49,00
107	48,68	47,79	48,17	48,80

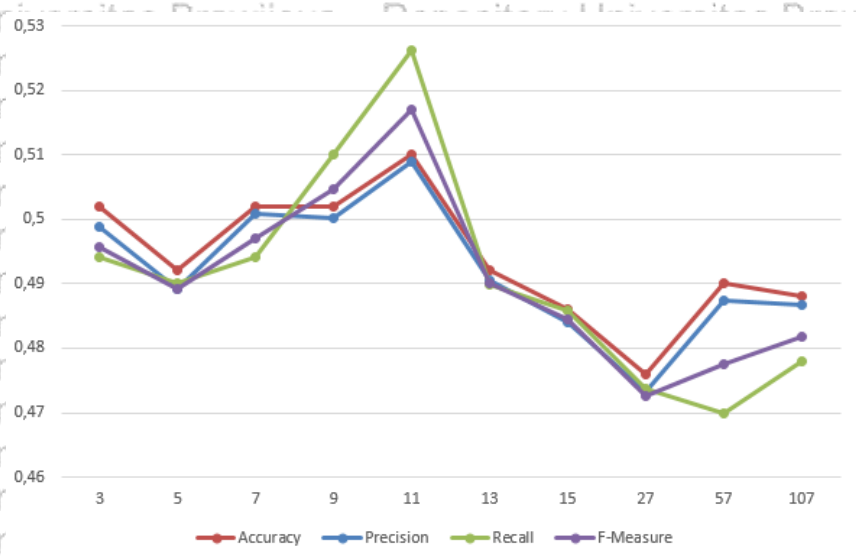
Berdasarkan tabel 6.6 dapat dilihat akurasi,presisi,*recall*, serta *f-measure* yang paling tinggi berada pada nilai *k*=11, yaitu 51,00%, 50,90%, 52,61%, dan 51,70%. Disimpulkan bahwa pada saat nilai *k*=11 mencapai nilai pengujian tertinggi.

6.3 Analisis

Sub bab ini akan menjabarkan hasil pengujian yang sudah diuji dengan 5-*fold cross validation.*

6.3.1 Analisis Pengujian Nilai K dan 5-*Fold Cross Validation*

Gambar 6.1 menampilkan grafik dari hasil nilai k yang telah diuji serta 5-*Fold Cross Validation.*



Gambar 6.1 Grafik Hasil Pengujian Nilai k dan 5-*Fold Cross Validation*

Gambar 6.1 menjelaskan hasil evaluasi (*f-measure, precision,* nilai akurasi dan *recall*) tertinggi terdapat pada k=11. Sedangkan nilai terendah berada pada nilai k=27. Hal tersebut menjelaskan bahwa nilai k yang semakin besar, tidak selalu menghasilkan nilai yang tinggi terhadap hasil evaluasi pengujian. Pada saat nilai k=3 hingga k=11 menunjukkan nilai *f-measure, precision,* nilai akurasi dan *recall* yang mengalami peningkatan, namun di nilai k=13 hingga k=107 justru *f-measure, precision,* nilai akurasi dan *recall* mengalami penurunan. Hal ini disebabkan karena nilai k berperan penting dalam proses klasifikasi hasil kelas, terdapat kondisi

dimana nilai k bernilai besar justru menghasilkan hasil evaluasi yang rendah, sedangkan nilai k yang rendah malah menghasilkan nilai evaluasi yang lebih tinggi. Hal ini disebabkan karena terdapat beberapa *term* yang bisa terklasifikasikan ke dalam kedua kelas sehingga mempengaruhi hasil evaluasi. Selain itu hasil evaluasi yang rendah disini juga dikarenakan terdapat term-term dalam bahasa asing. Tabel 6.7 dan tabel 6.8 menjelaskan contoh term-term tersebut.

Tabel 6.7 Contoh Term yang terklasifikasikan 2 kelas

Text	Class
produknya mantap, bikin muka mulus. walau awalnya emang rada panas, tapi lama kelamaan enggak. dingin di muka. bisa dijadiin sleeping mask + diolesin ke rambut	positif
Produk yang dicintai sejuta umat Tapi engga di aku, setelah beberapa bulan pake ini bikin jidat bruntusan	negatif
Sukak banget sama produk ini. Dari teksturnya enggak lengket di kulit, jadi langsung meresap. Dari wanginya wangi banget. Kalau yang asli itu bau alkohol sama wangi gitu	positif
Buat aku produk ini engga ada efeknya, saking keringnya muka aku apa ya? Aku pakai ini sebagai pelembab tapi engga meresap di kulitku berasa nempel aja	negatif

Dari Tabel 6.7 dapat dilihat contoh term ‘enggak’ yang terklasifikasikan pada kedua kelas. Sebenarnya term ‘enggak’ cenderung term yang negatif, tetapi disini tergantung dari kata yang menyertainya sehingga dapat terklasifikasikan pada kelas positif.

Tabel 6.8 Contoh Term yang terklasifikasikan 2 kelas

Text	Class
bener bener bisa ilangin bruntusan di muka, wajah jd glowing dan terlindungi	Positif
Aku makenya sbg pelembab dan sleeping mask dan menurutku biasa aja produknya..dipakenya enak sih berasa dingin, seger tapi kalo udah 2 jam pake ini doang mukaku yg oily jadi tambah keset kinclong silaw gitu, bukan glowing tapi ya silaw..dan gak ngebantu beruntusan juga	Negatif
tercintah number1 ini mah suka banget dong abis make ini wajah berasa glowing dan alus bisa jadi mouisturizer bisa juga jadi primer harganya murah untuk isi yang sebanyak	Positif
bagus sekali. mukaku benar benar breakout bruntusan. setiap malam aku pakein ini, jerawat kempes dan bekas hilang dalam waktu 3 bulan muka glowing. sayang formulanya mengandung	Negatif

alkohol, dan yaaa cukup bahaya sih. karna aku terkena spidervein akibat alkohol berlebih	

Selain itu dari Tabel 6.8 dapat dilihat salah satu term seperti term 'glowing' yang merupakan istilah asing. Seharusnya term 'glowing' cenderung ke dalam kelas positif, tetapi dapat terklasifikasikan ke dalam kelas negatif karena tergantung dari kata yang menyertainya sehingga terklasifikasikan ke dalam kelas negatif. Dikarenakan banyaknya *term term* seperti yang sudah dijelaskan, sangat berpengaruh pada hasil evaluasi pengujian.

BAB 7 PENUTUP

Bab ini menjelaskan penelitian yang disimpulkan penulis dan beberapa saran yang bisa digunakan sebagai pertimbangan guna penelitian selanjutnya yang bisa dikembangkan lebih baik lagi.

7.1 Kesimpulan

Kesimpulan dapat dijabarkan sebagai berikut:

1. Semakin besar nilai k tidak selalu menghasilkan nilai akurasi yang semakin tinggi pula. Hal tersebut dibuktikan dari hasil yang diuji menggunakan 5-*fold cross validation* yang menunjukkan pada saat nilai *k*=3 hingga *k*=11 menunjukkan nilai nilai Akurasi, *Precision, Recall,* dan *F-Measure* yang mengalami peningkatan, namun pada nilai *k*=13 hingga k=107 justru *f-measure, precision,* nilai akurasi dan *recall* mengalami penurunan.
2. Hasil akhir terbaik pada saat nilai *k*=11 dengan *f-measure, precision,* nilai akurasi dan *recall* sebesar 51,00%, 50,90%, 52,61%, dan 51,70%. Nilai yang tidak terlalu besar ini dipengaruhi oleh *term-term* yang dipakai pada data yang digunakan.

7.2 Saran

Saran oleh penulis yang nantinya dapat digunakan sebagai pertimbangan untuk mengembangkan penelitian selanjutnya dituangkan sebagai berikut:

1. Melakukan perbaikan pada *term-term* yang mengandung istilah asing serta *term-term* yang masih tidak baku karena hal tersebut mempengaruhi hasil evaluasi sehingga dokumen yang diuji dapat terklasifikasikan kedalam kelas yang sesuai dan dihasilkan hasil evaluasi yang lebih baik.
2. Menambahkan pengujian pada nilai *k* dan *b* sehingga dapat diketahui pengaruhnya dalam hasil evaluasi.
3. Menambahkan pengujian lain seperti pengujian seleksi fitur dengan *chi square*.

DAFTAR REFERENSI

Adiwijaya, I., 2006. *Text Mining dan Knowledge Discovery.* s.l., Kolokium bersama komunitas datamining Indonesia & Soft Computing Indonesia.

Binawan, D. H., I. & Adikara, P. P., 2019. Klasifikasi Dokumen Abstrak Skripsi Berdasarkan Fokus Penelitian di Bidang Komputasi Cerdas Menggunakan BM25 dan K-Nearest Neighbor. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer,* Volume 3, pp. 2640-2645.

Han, J., Kamber, M. & Pei, J., 2006. *Data Mining Concept and Techniques.* 3rd ed. Waltham, USA: Elsevier.

Kaur, H., Mangat, V. & N., 2017. *A Survey of Sentiment Analysis techniques.* Chandigarh, India, International conference on I-SMAC.

Kumalasari, N. A., M. & Dewi, C., 2014. IMPLEMENTASI ALGORITMA MODIFIED K-NEAREST NEIGHBOR (MKNN) UNTUK MENENTUKAN TINGKAT RESIKO PENYAKIT LEMAK DARAH (PROFIL LIPID). Volume 4.

O., M., Gazalba, I. & Indra Reza, N. G., 2017. *Comparative Analysis of K-Nearest Neighbor and Modified K-Nearest Neighbor Algorithm for Data Classification.* Pekanbaru, s.n., pp. 294-298.

Onantya, I. D., Indriati & Adikara, P. P., 2019. Analisis Sentimen Pada Ulasan Aplikasi BCA Mobile Menggunakan BM25 Dan Improved K-Nearest Neighbor. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer,* pp. 2575-2580.

Parvin, H., Alizadeh, H. & Bidgoli, B. M., 2008. *MKNN: Modified K-Nearest Neighbor.* San Francisco, WCECS.

PINANDHITA, R. R., 2013. *PERINGKAS DOKUMEN BERBAHASA INDONESIA BERBASIS KATA BENDA DENGAN BM25.* Bogor, repository ipb.

Prana, P. A., I. & Adikara, P. P., 2019. Klasifikasi Komentar Body Shaming Beauty Vlogger Pada Youtube Menggunakan Metode BM25 dan K-Nearest Neighbor. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer,* Volume 3, pp. 7328-7334.

Rachmat, A. & Delima, R., 2014. Implementasi Metode K-Nearest Neighbor dengan Decision Rule untuk Klasifikasi Subtopik Berita. *Jurnal Informatika,* pp. 1-15.

Royyan, A. N., I. & Muflikhah, L., 2018. Analisis Sentimen Review Aplikasi Mobile Dengan Menggunakan Metode Modified K Nearest Neighbour (MK-NN). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer,* pp. 3157-3162.

Sanjaya, F., 2017. Pemanfaatan Sistem Temu Kembali Informasi dalam Pencarian Dokumen Menggunakan Metode Vector Space Model. *Journal of Information and Technology,* Volume 05, pp. 147-153.

Sh, Z., Keung, J. & Song, Q., 2014. *An Empirical Study of BM25 and BM25F Based Feature*. Xi'an, IEEE.

Syifa, P., 2018. *Kompas.* [Online] Available at: https://ekonomi.kompas.com/read/2018/08/20/140853326/industri-kecantikan-di-indonesia-tumbuh-pesat-hingga-16-persen# [Accessed 22 10 2019].

Tinega, G. A., Mwangi, W. & Rimiru, R., 2018. Text Mining in DIgital Libraries using OKAPI BM25 Model. *International Journal of Computer Applications Technology and Research,* 7(10), pp. 398-406.

Wisnubrata, 2017. *lifestyle.kompas.com.* [Online] Available at: https://lifestyle.kompas.com/read/2017/06/14/135648020/apa.definisi.perempuan.cantik [Accessed 10 09 2019].

LAMPIRAN A DATASET

Lampiran 7.1 Dataset

No	Review	Kelas
1	nyaman untuk digunakan, dimuka lengket jadi sebisa mungkin tidak menyentuhnya. Membuat jerawat agak mengecil. Melembabkan muka juga. Dingin saat diaplikasikan di muka	negatif
2	Mosturizer andalan yg bisa multifungsi juga, karena bisa dipake buat bagian badan yg terasa kring juga. kalau malemnya pake ini, paginya muka jd halus dan lembab	positif
3	Udah setahun make produk ini dan masih jadi juara dikulit aku. Apalagi pas aku lihat kulit wajah temenku tambah bersih karena rutin pake produk ini. Teksturnya enak, bikin wajah dingin, pokoknya suka banget	positif
4	Ini hits banget booming dimanamana penasaran coba dan kurang suka sm ini ga Ada hasil yg bagus di mukaku trs lama jg nyerepnya , ga suka trs ninggalin residu jg	negatif
5	Dulu beli ini karena emang lagi hits banget kaan, semua orang pake dan orang-orang terdekat aku juga pada cocok	negatif
6	aloevera kecintaa segala umat yang sempat booming banget	positif
7	Holygrail hampir semua orang deh kayaknya ini. Ga melihat gender semua pakaiii. Efektif buat nenangin kulit dan melembabkan. Termasuk murah dengan jumlah isi segitu. Wanginya ga menyengat, teksturnya pas.	positif
8	awalnya aku beli ini karna tertarik banyak banget yang mereview yang super duper bagus tapi di aku baru banget cuman makai 1 kali aja soalnya ga cocok untuk type muka yang sensitiv dan berminyak	negatif
9	Beli ini akibat tergiur setelah baca dan nonton review orang yang bilang kalau ini holy grail-nya buat beberapa masalah kulit mereka. Sayangnya malah zonk di gue. Baru semalem tapi bruntusan gue malah nambah terus merah-merah. Mana banyak banget isinya huhu. Kayanya sih buat yg kulitnya sensitif harus hati-hati karena alkohol dan fragrance-nya juga kenceng	negatif
10	bener bener bisa ilangin bruntusan di muka, wajah jd glowing dan terlindungi	positif
11	sedih sih produk hype yang satu ini malah bikin kulit aku kering banget!! huhu padahal aku pikir produk ini bisa ngehidrasi kulit aku yang kering ini tapi ternyata makin bikin kering... sedih deh	negatif

12	bikin adem banget apalagi kalau wajah kita lagi capek banget	positif
13	Melembabkan hanya sementara, jika kena luka suka ada rasa perih perih, mungkin bisa dipakai campuran masker, di mukaku malah bikin kering kalau dipake moisturizer	negatif
14	LOVE BANGET. Salah satu produk yang ngebantu bruntusan dan ngebantu banget kalau wajah aku lagi sun burn gitu. Apalagi kalau dimasukin ke freezer	positif
15	bagus parah buat yang lagi beruntusan, cepet kempesnya, nyerahin juga, ngeratain tekstur	negatif
16	Aku udah beberapa kali pakai ini dan lembab banget ke kulit untuk sebagai sleeping mask dan paginya waah lembab sekali	positif
17	produk yang katamya bagus tapi entah kenapa di aku gak cocok sama sekali. bikin wajah aku jadi jerawatan padahal aku beli ini ori di counter NR nya langsung	negatif
18	Sleeping mask favorit aku ini, paginya berasa seger dan kenyal. Buat primer jg oke Buat rambut mantap jg	positif
19	bagus banget untuk wajah yg sering beruntusan! aku selalu pakai ini jadi moisturizer, campuran masker, ataupun sleeping mask. secocok itu emang 1 tahun terakhir pakai ini natrep ini juga bisa dipakai kompres mata kalo lagi capek banget luv pokoknya!	positif
	……..	
500	gak suka karena abis make ini aku tumbuh jerawat. sangat tidak di rekomendasikan untuk kulit sensi karena ada alkoholnya	negatif

Keterangan:

Dataset lengkat dapat dilihat pada link berikut ini bit.ly/DatasetAlfita