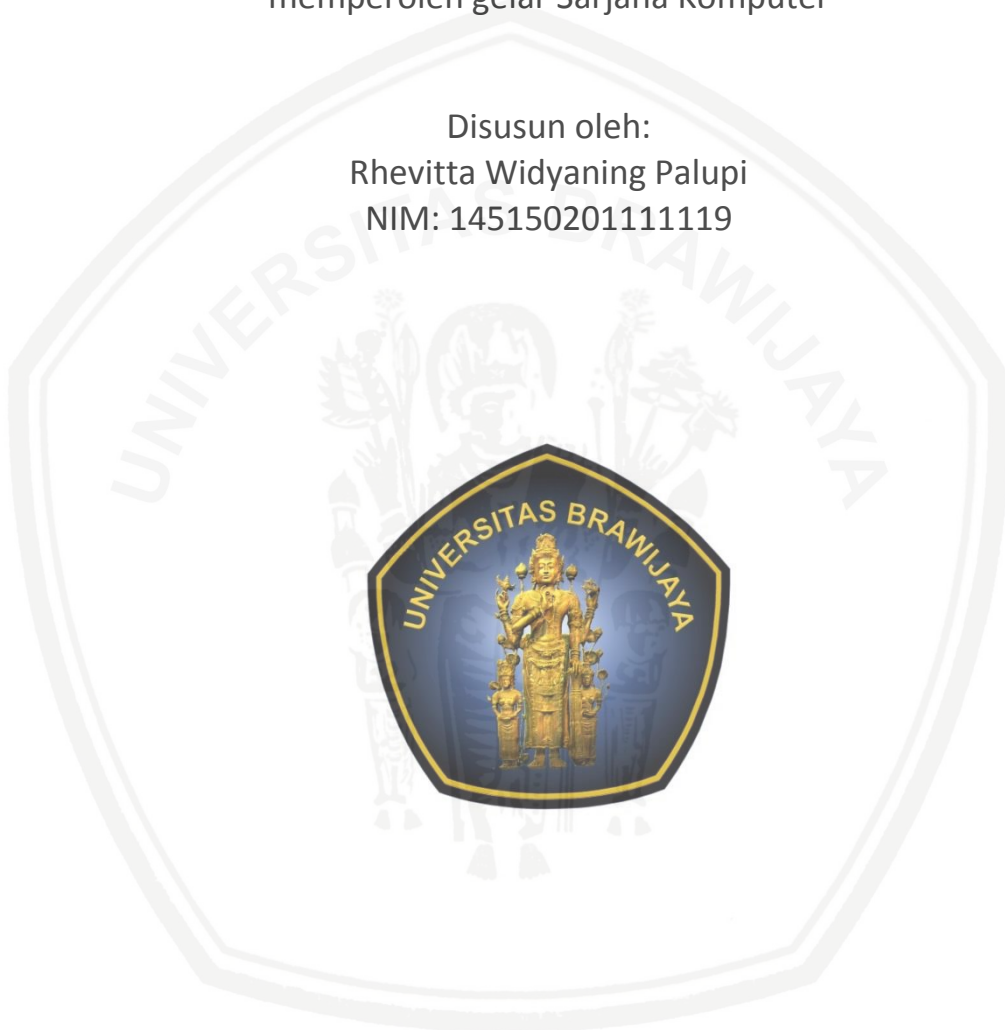


**PREDIKSI RATING NOVEL BARU BERDASARKAN SINOPSIS
MENGUNAKAN *GENRE BASED COLLABORATIVE FILTERING*
DAN *TEXT SIMILARITY***

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Rhevitta Widyaning Palupi
NIM: 145150201111119



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018**

PENGESAHAN

PREDIKSI RATING NOVEL BARU BERDASARKAN SINOPSIS MENGGUNAKAN *GENRE*
BASED COLLABORATIVE FILTERING DAN *TEXT SIMILARITY*

SKRIPSI

Untuk memenuhi sebagian persyaratan
Memperoleh gelar Sarjana Komputer

Disusun Oleh:

Rhevitta Widyaning Palupi

NIM: 145150201111119

Skripsi ini telah diuji dan dinyatakan lulus pada
27 Desember 2018

Telah diperiksa dan disetujui oleh:

Pembimbing I

Pembimbing II

Yuita Arum Sari, S.Kom., M.Kom
NIK: 201609 880715 2 001

Putra Pandu Adikara, S.Kom., M.Kom.
NIP: 19850725 200812 1 002

Mengetahui

Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T., M.T., Ph.D.
NIP: 19710518 200312 1 001



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 memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata di 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 Pasal 70).

Malang, 17 Desember 2018



Rhevitta Widyaning Palupi

NIM: 145150201111119

PRAKATA

Alhamdulillah rabbi'alamin. Puji dan syukur penulis ucapkan atas kehadiran Allah SWT atas rahmat dan karunia-Nya kepada penulis, sehingga dapat menyelesaikan skripsi yang berjudul "Prediksi Rating Novel Baru Berdasarkan Sinopsis Menggunakan *Genre Based Collaborative Filtering* dan *Text Similarity*" dengan baik.

Dalam pengerjaan skripsi ini penulis mendapatkan banyak dukungan dan bantuan baik moral maupun material dari pelbagai pihak. Oleh karena itu penulis mengucapkan banyak terima kasih kepada:

1. Ibu Yuita Arum Sari, S.Kom, M.Kom dan Bapak Putra Pandu Adikara, S.Kom, M.Kom selaku dosen pembimbing yang telah sabar dan memberi banyak saran dan pengarahan untuk skripsi penulis sehingga penulis dapat menyelesaikan skripsi ini.
2. Bapak Agus Wahyu Widodo, S.T., M.Cs selaku Ketua Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya Malang.
3. Bapak M. Tanzil Furqon, S.Kom., M.Comp.Sc selaku Sekretaris Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya Malang.
4. Bapak Tri Astoto Kurniawan, S.T., M.T., Ph.D selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya Malang.
5. Kedua orang tua dan saudara penulis, ayah dan ibu yang saya hormati dan saya cintai Bapak Asrori dan Ibu Musringah S.Pd.SD., mas Denny, mas Wahyu Aji, dan adik Hamdan yang selalu memberikan dukungan baik moriil maupun materiil, motivasi, semangat, serta doa demi kelancaran dalam pengerjaan skripsi ini.
6. Seluruh Civitas Akademika Fakultas Ilmu Komputer Universitas Brawijaya yang telah memberikan bantuan dan dukungan selama penulis menempuh studi dan menyelesaikan skripsi di Fakultas Ilmu Komputer Universitas Brawijaya.
7. Teman-teman yang saling mendukung dan membantu penulis baik secara langsung maupun tidak langsung yaitu Dayinta, Ni'mah, Via Aprilia, Amalia, Firdaus, Dwi Wahyu, Fathony, dan yang lainnya.
8. Teman-teman Kos Sumpersari 254 yang selalu mendukung dan mendoakan kelancaran skripsi penulis.
9. Seluruh pihak yang telah membantu kelancaran dalam pengerjaan skripsi yang tidak dapat penulis sebutkan satu per satu.

Semoga Allah SWT senantiasa membalas segala kebaikan dan jasa yang telah diberikan kepada penulis. Pada intinya, skripsi ini membasa mengenai sistem yang mampu memberikan prediksi rating dengan menggunakan algoritme *Genre Based Collaborative Filtering* dan *Text Similarity* pada novel baru berdasarkan sinopsis.

Penulis menyadari bahwa skripsi ini memiliki banyak kekurangan baik pada format penulisan maupun isi bahasannya, oleh karena itu diharapkan kritik yang dapat membangun. Semoga skripsi ini dapat memberikan manfaat bagi semua pihak baik untuk pembaca maupun penelitian selanjutnya.

Malang, 17 Desember 2018

Penulis

rhevittawidyaningpalupi@gmail.com



ABSTRAK

Rhevitta Widyaning Palupi, *Prediksi Rating Novel Baru Berdasarkan Sinopsis Menggunakan Genre Based Collaborative Filtering dan Text Similarity*

Dosen Pembimbing: Yuita Arum Sari, S.Kom., M.Kom. dan Putra Pandu Adikara, S.Kom., M.Kom.

Novel merupakan suatu cerita yang memiliki alur panjang yang bersifat imajiantif. Berdasarkan pilihan editor pada situs Amazon.com, 50 dari 100 buku dengan penjualan terbaik merupakan novel. Hal tersebut menunjukkan bahwa ketertarikan masyarakat terhadap novel cukup tinggi sebagai salah satu jenis bacaan. Namun saat ingin memilih novel yang hendak dibaca, pembaca terkadang merasa bingung untuk mengetahui kualitas dari novel tersebut. Salah satu acuan dalam melihat kualitas suatu produk yaitu rating. Situs Goodreads merupakan salah satu situs yang memungkinkan peninjau amatir menuliskan ulasan serta rating untuk membantu pembaca dalam memilih buku yang relevan. Namun terkadang pengguna Goodreads tidak memberikan rating terhadap suatu buku sehingga pengikut dari pengguna tersebut ingin mengetahui rating yang diberikan pengguna pada buku tersebut. Penelitian ini menggunakan metode *Genre Based Collaborative Filtering* sebagai penghitungan prediksi rating dan *Text Similarity* untuk mengetahui nilai kesamaan antara dokumen yang satu dengan lainnya. Data yang digunakan pada penelitian ini sebanyak 31 user dan 90 sinopsis sebagai data latih dan 35 sinopsis sebagai data uji. Akurasi sistem yang diperoleh dari hasil klasifikasi dengan menggunakan nilai kesamaan pada *text similarity* sebesar 45,714286% dan nilai MAE sebesar 0,27742857 sehingga dapat disimpulkan bahwa metode *genre based collaborative filtering* dan *text similarity* dapat digunakan untuk melakukan prediksi rating.

Kata kunci: *prediksi rating, novel, Goodreads, Genre Based Collaborative Filtering, Text Similarity*

ABSTRACT

Rhevitta Widyaning Palupi, Prediction of Rating New Novel Based on Synopsis Using Genre Based Collaborative Filtering and Text Similarity

Advisor: Yuita Arum Sari, S.Kom., M.Kom. and Putra Pandu Adikara, S.Kom., M.Kom.

The novel is a story that has a long, imaginary plot. Based on the editor's choice on the Amazon.com website, 50 of the 100 best-selling books are novels. This shows that public interest in the novel is quite high as one type of reading. But when you want to choose a novel that you want to read, readers sometimes feel confused to know the quality of the novel. One reference in looking at the quality of a product is rating. The Goodreads site is one site that allows amateur reviewers to write reviews and ratings to help readers choose relevant books. But sometimes Goodreads users don't give ratings to a book so followers from that user want to know the rating given by the user in the book. This study uses the Genre Based Collaborative Filtering method as a calculation of rating predictions and Text Similarity to determine the value of similarity between documents with each other. The data used in this study were 31 users and 90 synopsis as training data and 35 synopsis as test data. System accuracy obtained from the classification results by using the similarity value on text similarity of 45,714286% and MAE value of 0,27742857 so that it can be concluded that the method of genre based collaborative filtering and text similarity can be used to make rating predictions.

Keywords: *rating predictions, novels, Goodreads, Genre Based Collaborative Filtering, Text Similarity*

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
PRAKATA.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xii
DAFTAR KODE PRPOGRAM	xiii
DAFTAR LAMPIRAN	xiv
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	3
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan masalah	3
1.6 Sistematika pembahasan.....	4
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Novel	8
2.3 Collaborative Filtering.....	8
2.3.1 Item Based Collaborative Filtering.....	8
2.4 Text Similarity	9
2.4.1 Tokenization	9
2.4.2 Cleaning.....	10
2.4.3 Case Folding	10
2.4.4 Filtering	10
2.4.5 Stemming	11
2.4.6 TF-IDF	11
2.4.7 Cosine Similarity.....	11
2.5 Accuracy.....	12

2.6 Mean Absolute Error (MAE).....	12
BAB 3 METODOLOGI PENELITIAN	13
3.1 Tipe Penelitian	13
3.2 Strategi Penelitian.....	13
3.3 Partisipan Penelitian	13
3.4 Lokasi Penelitian	13
3.5 Teknik Pengumpulan Data	13
3.6 Implementasi Algoritme	15
3.7 Teknik Analisis Data	16
3.8 Kesimpulan dan Saran	16
BAB 4 PERANCANGAN.....	17
4.1 Deskripsi permasalahan.....	17
4.2 Deskripsi umum sistem.....	17
4.3 <i>Genre Based Collaborative Filtering</i>	18
4.4 <i>Text Similarity</i>	21
4.4.1 <i>Tokenization</i>	22
4.4.2 <i>Cleaning</i>	23
4.4.3 <i>Case Folding</i>	24
4.4.4 <i>Filtering</i>	25
4.4.5 <i>Stemming</i>	27
4.4.6 TF-IDF	28
4.4.7 <i>Cosine Similarity</i>	30
4.5 Perhitungan Manual	32
4.6 Perancangan Pengujian	46
BAB 5 IMPLEMENTASI	47
5.1 Implementasi Sistem	47
5.1.1 <i>Genre Based Collaborative Filtering</i>	47
5.1.2 <i>Tokenization</i>	48
5.1.3 <i>Cleaning</i>	48
5.1.4 <i>Case Folding</i>	49
5.1.5 <i>Filtering</i>	50
5.1.6 <i>Stemming</i>	51

5.1.7 TF-IDF	52
5.1.8 <i>Cosine Similarity</i>	53
BAB 6 HASIL DAN PEMBAHASAN	55
6.1 Hasil dan Analisis Prediksi Rating.....	55
6.2 Hasil dan Analisis Klasifikasi Sinopsis.....	58
BAB 7 PENUTUP	61
7.1 Kesimpulan.....	61
7.2 Saran	61
DAFTAR REFERENSI	62
LAMPIRAN	64



DAFTAR TABEL

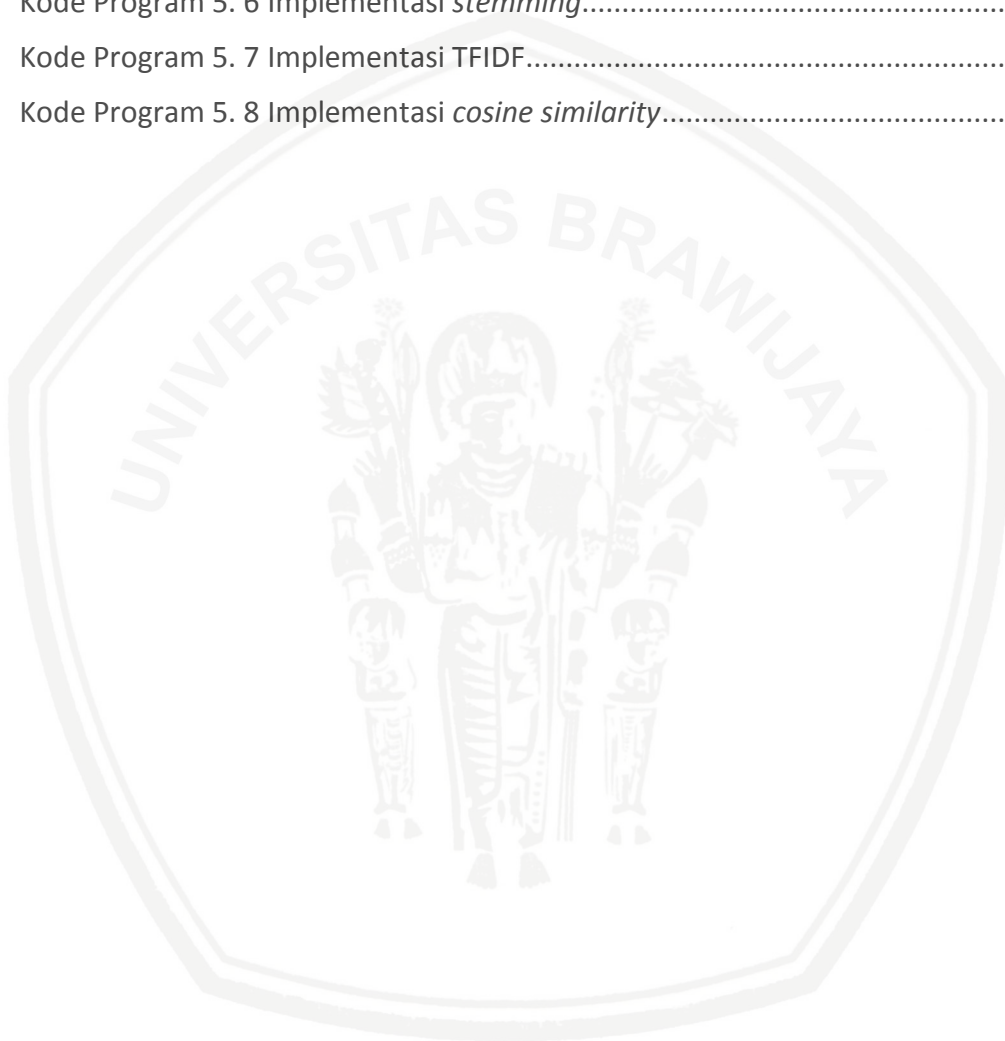
Tabel 2. 1 Penelitian sebelumnya	6
Tabel 2. 2 Contoh <i>tokenization</i> teks dokumen Bahasa Inggris	9
Tabel 2. 3 Contoh <i>cleaning</i> teks dokumen Bahasa Inggris.....	10
Tabel 2. 4 Contoh <i>case folding</i> teks dokumen Bahasa Inggris.....	10
Tabel 2. 5 Contoh <i>filtering</i> pada dokumen teks Bahasa Inggris.....	10
Tabel 2. 6 Contoh <i>stemming</i> pada teks dokumen Bahasa Inggris	11
Tabel 4. 1 Data matriks <i>user-genre rating</i>	32
Tabel 4. 2 Nilai rata-rata <i>user</i>	33
Tabel 4. 3 Matriks kesamaan <i>genre-to-genre</i>	33
Tabel 4. 4 Matriks <i>user-genre rating</i>	33
Tabel 4. 5 Hasil prediksi rating	34
Tabel 4. 6 Dokumen sinopsis novel.....	34
Tabel 4. 7 Manualisasi <i>Tokenization</i>	35
Tabel 4. 8 Manualisasi <i>Cleaning</i>	37
Tabel 4. 9 Manualisasi <i>Case Folding</i>	39
Tabel 4. 10 Manualisasi <i>Filtering</i>	41
Tabel 4. 11 Manualisasi <i>Stemming</i>	42
Tabel 4. 12 Manualisasi <i>Binary TF</i> dan <i>Raw TF</i>	43
Tabel 4. 13 Manualisasi <i>Log TF</i> dan <i>IDF</i>	44
Tabel 4. 14 Manualisasi <i>TF-IDF</i>	44
Tabel 4. 15 Manualisasi <i>Cosine Similarity</i>	45
Tabel 4. 16 Tabel perancangan pengujian	46
Tabel 6. 1 Hasil prediksi rating oleh user untuk masing-masing sinopsis	55
Tabel 6. 2 Hasil rata-rata rating per genre	55
Tabel 6. 3 Hasil pengujian sistem	56
Tabel 6. 4 Hasil analisis pengujian.....	58

DAFTAR GAMBAR

Gambar 3. 1 Pengambilan data berupa rating, genre, serta sinopsis	14
Gambar 3. 2 Pengambilan rating yang diberikan oleh <i>user</i>	15
Gambar 3. 3 Perancangan Algoritme	15
Gambar 4. 1 Deskripsi umum sistem	18
Gambar 4. 2 Alir proses <i>genre based collaborative filtering</i>	20
Gambar 4. 3 Diagram alir <i>pre-processing</i>	21
Gambar 4. 4 Diagram alir proses <i>tokenization</i>	22
Gambar 4. 5 Diagram alir proses <i>cleaning</i>	24
Gambar 4. 6 Diagram alir <i>case folding</i>	25
Gambar 4. 7 Diagram alir <i>filtering</i>	26
Gambar 4. 8 Diagram alir <i>stemming</i>	27
Gambar 4. 9 Diagram alir proses TF-IDF	29
Gambar 4. 10 Diagram alir proses <i>Cosine Similarity</i>	31
Gambar 6. 1 Grafik pengujian rating asli dengan rating prediksi	58

DAFTAR KODE PRPROGRAM

Kode Program 5. 1 Prediksi rating dengan <i>genre based collaborative filtering</i> ...	48
Kode Program 5. 2 Implementasi <i>tokenization</i>	48
Kode Program 5. 3 Implementasi <i>cleaning</i>	49
Kode Program 5. 4 Implementasi <i>case folding</i>	50
Kode Program 5. 5 Implementasi <i>Filtering</i>	50
Kode Program 5. 6 Implementasi <i>stemming</i>	51
Kode Program 5. 7 Implementasi TFIDF.....	53
Kode Program 5. 8 Implementasi <i>cosine similarity</i>	54



DAFTAR LAMPIRAN

Lampiran 1 Tabel Data Awal Rating Oleh User	64
Lampiran 2 Tabel Data Latih Sinopsis	64
Lampiran 3 Tabel Hasil Prediksi Rating User pada Semua Sinopsis.....	66
Lampiran 4 Tabel Data Uji	67



BAB 1 PENDAHULUAN

1.1 Latar belakang

Novel adalah suatu cerita yang memiliki alur panjang yang bersifat imajinatif, sehingga novel dapat dikatakan sebagai suatu karya fiksi berupa prosa yang menceritakan tokoh-tokoh dalam suatu alur yang panjang (Tarigan, 2011). Kata novel diadopsi dari bahasa Italia yaitu *novella* yang secara harfiah maksudnya yaitu barang baru yang berbentuk kecil yang selanjutnya diartikan sebagai suatu bentuk prosa (Nurgiyantoro, 1995). Novel merupakan salah satu jenis buku bacaan yang diminati oleh masyarakat. ISBN mencatat terdapat 50.498 buku fiksi baru telah terbit di tahun 2013 (ISBN, 2014). Berdasarkan pilihan editor pada situs Amazon.com menunjukkan bahwa 50 dari 100 buku dengan penjualan terbaik sepanjang masa merupakan novel (Amazon.com, 2018). Hal tersebut membuktikan tingginya ketertarikan masyarakat terhadap novel sebagai salah satu jenis bacaan. Namun ketika ingin membaca novel baru, pembaca terkadang merasa kebingungan untuk menentukan dan mengetahui kualitas dari novel tersebut.

Rating merupakan salah satu bagian dari *review* yang menggunakan simbol bintang dalam mengekspresikan pendapat dari pelanggan. Rating dapat diartikan sebagai penilaian pengguna pada suatu produk tertentu berdasarkan pengalaman pengguna saat menggunakan dengan produk tersebut (Li & Zhang, 2002). Dikutip dari advertising-indonesia.id rating digunakan sebagai acuan dalam melihat kualitas suatu program atau produk berdasarkan respon masyarakat terhadap program atau produk tersebut (Advertising-Indonesia, 2017). Salah satu situs terbesar di dunia yang fokus pada pembaca dan rekomendasi buku yaitu goodreads.com. Situs Goodreads memiliki delapan puluh juta anggota, dua juta tiga ratus ribu buku, serta memiliki delapan puluh juta *review* atau ulasan (Goodreads, 2018). Situs Goodreads memungkinkan peninjau amatir untuk menuliskan ulasan tentang buku serta memberikan kritik pedas kepada penulis novel karena tinjauan dari profesional cenderung positif (Jane, 2014). Selain ulasan, Goodreads juga memberikan fasilitas rekomendasi dan rating yang dapat membantu pembaca untuk memilih buku yang relevan (Thelwall & Kousha, 2016). Namun terkadang pengguna Goodreads hanya membaca tanpa memberikan rating sehingga pengikut dari pengguna tersebut ingin tahu berapa rating yang diberikan oleh pengguna tersebut pada suatu novel tertentu.

Ming-jia Wang dan Jin-ti Han melakukan penelitian dengan menggunakan algoritme *item based collaborative filtering*. Metode pada penelitian ini juga digunakan untuk menghitung nilai rating dari item yang belum dinilai oleh pengguna. Hasil dari penelitian ini menunjukkan bahwa algoritme *collaborative filtering* yang mengambil informasi karakteristik pada *item* dapat meningkatkan akurasi prediksi dan rekomendasi (Wang & Han, 2012). Selain itu, Puntheeranurak dan Chaiwitooanukool melakukan penelitian mengenai penggunaan *item based hybrid similarity* pada metode *item based collaborative filtering*. *Item based*

collaborative filtering sering digunakan dalam sistem rekomendasi. Sistem rekomendasi menggunakan nilai *similarity rating item* untuk memprediksi hal yang disukai pengguna. Puntheeranurak dan Chaiwitooanukool memasukkan nilai *similarity attribute item* untuk menyesuaikan persamaan rating yang diprediksi dengan item target. Nilai Mean Absolute Error (MAE) yang dihasilkan *item based hybrid similarity* dan teknik *similarity attribute item* lebih sedikit daripada teknik *item based collaborative filtering* yang tradisional (Puntheeranurak & Chaiwitooanukool, 2011).

Ye Zhang dan Wei Song menemukan permasalahan apabila salah satu dari dua *item* diberikan rating yang sangat tinggi dan yang lainnya sangat rendah meskipun kedua *item* tersebut memiliki genre yang sama dengan *item* target, maka tidak bisa diperkirakan yang memiliki kesamaan yang tinggi dengan *item* target. Sehingga Ye Zhang dan Wei Song melakukan penelitian mengenai *item based collaborative filtering* menggunakan kemiripan genre. Apabila kelompok pengguna memberikan rating yang tinggi pada *item* target dan *item* lain yang memiliki genre yang sama, maka dapat dikatakan bahwa *item* target dan *item* lain hampir sama. Algoritme *collaborative filtering* berdasarkan genre *item* dan kemiripan rating menghasilkan nilai Mean Absolute Error (MAE) yang lebih rendah daripada *item based* dan *user based* (Zhang & Song, 2009).

Alexandre dan Michel melakukan penelitian mengenai rekomendasi profil pada *social website* dengan menggabungkan antara *collaborative filtering* dengan *text similarity*. Teknik pada *text similarity* memiliki hasil *precision* yang lebih baik pada tingkat *recall* yang lebih kecil, sedangkan pada *collaborative filtering* memberikan *precision* yang lebih baik pada tingkat *recall* yang lebih tinggi (Spaeth & Desmarais, 2013). Selain itu Bening melakukan penelitian mengenai klasifikasi berita *online* dengan menggunakan pembobotan TF-IDF dan *Cosine Similarity*. Untuk melakukan proses tersebut dilakukan proses *preprocessing* diantaranya tokenisasi, penghapusan *stopword*, dan *stemming* untuk memperkecil term sehingga mempercepat proses perhitungan pembobotan. Klasifikasi yang dilakukan oleh Bening memiliki tingkat akurasi sebesar 91,25% (Herwijayanti, et al., 2018).

Dari penjelasan tersebut maka pada penelitian ini akan dilakukan prediksi rating pada novel baru berdasarkan sinopsis menggunakan *genre based collaborative filtering* dan *text similarity*. Diharapkan dengan menggunakan metode tersebut, pembaca novel dapat mendapatkan informasi mengenai kualitas buku yang hendak dibaca berdasarkan rating yang telah diprediksi. Oleh karena itu, maka dilakukan penelitian dengan judul "Prediksi Rating Novel Baru Berdasarkan Sinopsis Menggunakan *Genre Based Collaborative Filtering* Dan *Text Similarity*", dimana penelitian ini ditujukan untuk melakukan prediksi rating pada novel baru dengan menggunakan *genre based collaborative filtering* dan *text similarity*.

1.2 Rumusan masalah

Berdasarkan latar belakang yang telah dijelaskan, maka rumusan masalah penelitian yang diambil adalah sebagai berikut;

1. Bagaimana merancang *Genre Based Collaborative Filtering* dan *Text Similarity* pada sinopsis novel untuk melakukan prediksi rating pada novel baru?
2. Bagaimana evaluasi yang dihasilkan *Genre Based Collaborative Filtering* dan *Text Similarity* pada sinopsis novel untuk melakukan prediksi rating pada novel baru?

1.3 Tujuan

Tujuan penyusunan skripsi ini adalah;

1. Melakukan prediksi rating pada novel baru berdasarkan sinopsis menggunakan *Genre Based Collaborative Filtering* dan *Text Similarity*.
2. Melakukan evaluasi yang dihasilkan *Genre Based Collaborative Filtering* dan *Text Similarity* pada sinopsis novel untuk melakukan prediksi rating pada novel baru.

1.4 Manfaat

Manfaat yang didapatkan dari penelitian yang dilakukan antara lain:

1. Bagi pengguna
Diharapkan pengguna dapat mengetahui prediksi rating pada novel baru berdasarkan sinopsis novel tersebut.
2. Bagi pembaca
Dari hasil penelitian yang didapatkan diharapkan pembaca dapat menambah wawasan mengenai prediksi dengan menggunakan metode *Genre Based Collaborative Filtering* dan *Text Similarity*.
3. Bagi penulis
Menjadi pembelajaran dan mampu mengimplementasikan ilmu yang telah dipelajari serta mendapatkan wawasan dan pengetahuan mengenai *Collaborative Filtering* dan *Text Similarity*.

1.5 Batasan masalah

Batasan masalah yang digunakan sebagai pedoman dan pelaksanaan penelitian ini adalah:

1. Dokumen informasi yang digunakan sebanyak 91 sinopsis dan 31 user aktif yang didapat dari goodreads.com
2. Dokumen informasi menggunakan teks berbahasa Inggris
3. Metode yang digunakan yaitu *Genre Based Collaborative Filtering* dan *Text Similarity*

1.6 Sistematika pembahasan

Penyusunan skripsi ini berdasarkan sistematika sebagai berikut:

BAB I PENDAHULUAN

Berisi mengenai latar belakang dari masalah, rumusan masalah, tujuan, manfaat dari penelitian, batasan masalah, serta sistematika pembahasan.

BAB II LANDASAN KEPUSTAKAAN

Berisi uraian mengenai konsep dasar yang dilakukan dalam penelitian yang digunakan pada penelitian sebelumnya, teori-teori yang digunakan dalam penelitian, metode, serta kepastakaan yang berhubungan dengan novel, *Genre Based Collaborative Filtering* dan *Text Similarity*.

BAB III METODOLOGI PENELITIAN

Memuat uraian mengenai metodologi yang akan digunakan pada penelitian diantaranya yaitu tipe penelitian yang diterapkan, strategi penelitian, partisipan penelitian untuk mendukung penelitian, lokasi penelitian, teknik pengumpulan data, implementasi algoritme yang digunakan dalam penelitian, teknik analisis data, serta kesimpulan dan saran.

BAB IV PERANCANGAN

Berisi uraian mengenai manualisasi, perancangan algoritme, dan perancangan pengujian yang diperlukan untuk melakukan prediksi rating menggunakan *Genre Based Collaborative Filtering* dan *Text Similarity*.

BAB V IMPLEMENTASI

Memuat pembahasan implementasi dalam melakukan penerapan *Genre Based Collaborative Filtering* dan *Text Similarity*.

BAB VI HASIL DAN PEMBAHASAN

Memuat pembahasan hasil pengujian dalam melakukan prediksi rating menggunakan *Genre Based Collaborative Filtering* dan *Text Similarity*.

BAB VII PENUTUP

Berisi mengenai kesimpulan dan saran dari penelitian yang telah dilakukan. Saran diberikan untuk pengembangan penelitian selanjutnya.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Kajian pustaka memuat penelitian-penelitian yang telah dilakukan sebelumnya, diantaranya penelitian yang dilakukan oleh Ming-jia Wang yaitu menggunakan algoritme *collaborative filtering* berdasarkan pada *item rating*. Metode ini dapat menghitung peringkat *item* proyek yang belum diberi peringkat berdasarkan analisis informasi karakteristik *item*, dan menggunakan *collaborative filtering* berdasarkan *item* untuk menemukan *item* yang serupa. Selain itu, metode pada penelitian ini juga digunakan untuk menghitung nilai rating dari *item* yang belum dinilai oleh pengguna. Hasil dari penelitian ini menunjukkan bahwa algoritme *collaborative filtering* yang mengambil informasi karakteristik pada *item* dapat meningkatkan akurasi prediksi dan rekomendasi (Wang & Han, 2012).

Kemudian terdapat penelitian yang dilakukan oleh Puntheeranurak dan Chaiwitooanukool melakukan penelitian mengenai penggunaan *item based hybrid similarity* pada metode *item based collaborative filtering*. *Item based collaborative filtering* sering digunakan dalam sistem rekomendasi. Sistem rekomendasi menggunakan nilai *similarity rating item* untuk memprediksi hal yang disukai pengguna. Puntheeranurak dan Chaiwitooanukool memasukkan nilai *similarity attribute item* untuk menyesuaikan persamaan rating yang diprediksi dengan *item target*. Nilai Mean Absolute Error (MAE) yang dihasilkan *item based hybrid similarity* dan teknik *similarity attribute item* lebih sedikit daripada teknik *item based collaborative filtering* yang tradisional (Puntheeranurak & Chaiwitooanukool, 2011).

Ye Zhang dan Wei Song menemukan permasalahan apabila salah satu dari dua *item* diberikan rating yang sangat tinggi dan yang lainnya sangat rendah meskipun kedua *item* tersebut memiliki genre yang sama dengan *item target*, maka tidak bisa diperkirakan yang memiliki kesamaan yang tinggi dengan *item target*. Sehingga Ye Zhang dan Wei Song melakukan penelitian mengenai *item based collaborative filtering* menggunakan kemiripan genre. Apabila kelompok pengguna memberikan rating yang tinggi pada *item target* dan *item* lain yang memiliki genre yang sama, maka dapat dikatakan bahwa *item target* dan *item* lain hampir sama. Algoritme *collaborative filtering* berdasarkan genre *item* dan kemiripan rating menghasilkan nilai Mean Absolute Error (MAE) yang lebih rendah daripada *item based* dan *user based* (Zhang & Song, 2009).

Penelitian yang dilakukan oleh Alexandre untuk merekomendasikan profil pada *social website* dengan menggabungkan antara *collaborative filtering* dengan *text similarity*. Teknik pada *text similarity* memiliki hasil *precision* yang lebih baik pada tingkat *recall* yang lebih kecil, sedangkan pada *collaborative filtering* memberikan *precision* yang lebih baik pada tingkat *recall* yang lebih tinggi (Spaeth & Desmarais, 2013).

Tabel 2. 1 Penelitian sebelumnya

No	Judul	Obyek	Metode	Hasil
1	<i>Collaborative Filtering Recommendation Based on Item Rating and Characteristic Information Prediction</i>	100000 rating pada 1682 film oleh 943 user, masing-masing user memberikan rating pada 20 film	<i>Item-based collaborative filtering</i>	<i>Collaborative filtering</i> dengan mengambil informasi karakteristik pada item dapat meningkatkan akurasi prediksi dan rekomendasi
2	<i>An Item-based Collaborative Filtering Method using Item-based Hybrid Similarity</i>	100000 rating pada 1682 film oleh 943 user, masing-masing user memberikan rating pada 20 film	<i>Item-based hybrid similarity</i> pada metode <i>item based collaborative filtering</i> dan teknik kesamaan rating item	Nilai MAE dari <i>Item-based hybrid similarity</i> pada metode <i>item based collaborative filtering</i> dan teknik kesamaan rating item lebih kecil daripada <i>item-based collaborative filtering</i> tradisional
3	<i>A Collaborative Filtering Recommendation Algorithm Based on Item Genre and Rating Similarity</i>	10000 rating pada 1682 film oleh 943 user	Algoritme <i>collaborative filtering</i> berdasarkan genre item dan kemiripan rating	<i>Improved item genre</i> dan kemiripan rating menghasilkan nilai MAE paling kecil, kemudian disusul <i>item based</i> dengan menggunakan genre, <i>item-based</i> , dan <i>user-based</i> yang memiliki nilai MAE paling tinggi
4	<i>Combining Collaborative Filtering dan Text Similarity for Expert</i>	Informasi profil pada <i>Social Website</i>	Kombinasi <i>tag</i> dan <i>text similarity</i> , kombinasi <i>text similarity</i> dan <i>tag</i>	Kombinasi <i>tag</i> dan <i>text similarity</i> memiliki tingkat <i>recall</i> antara 0.05 dan 0.4.

No	Judul	Obyek	Metode	Hasil
	<i>Profile Recommendations in Social Websites</i>		<i>recommendation dengan classical collaborative filtering</i>	
5	Klasifikasi Berita <i>Online</i> dengan menggunakan Pembobotan TF-IDF dan <i>Cosine Similarity</i>	Berita yang diambil dari <i>website</i> berita <i>online</i> yaitu kompas.com	Metode yang digunakan yaitu TF-IDF sebagai pernghitung bobot term dan cosine similarity untuk melihat kedekatan antar dokumen	Percobaan pengujian dilakukan empat kali dan dari empat kali pengujian dengan rata-rata akurasi sebesar 91,25% dan diketahui bahwa semakin banyak data uji maka akan menghasilkan tingkat akurasi yang semakin tinggi
6	Penerapan Algoritma <i>Cosine Similarity</i> dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi	Menggunakan 50 dokumen skripsi sebagai data latih dari beberapa kategori yang berbeda	Algoritme yang digunakan yaitu cosine similarity yang digunakan untuk klasterisasi dengan TF-IDF sebagai penghitung bobot termnya	Pengujian penelitian ini menghasilkan akurasi sistem sebesar 88,3% yang mana 49 dari 50 dokumen skripsi dapat diklasifikasikan dengan benar, dan hanya satu dokumen yang hasil klasifikasinya yang salah.

2.2 Novel

Kata novel diadopsi dari bahasa Italia yaitu *novella* yang secara harfiah maksudnya yaitu barang baru yang berbebtuk kecil yang selanjutnya diartikan sebagai suatu bentuk prosa (Nurgiyantoro, 1995). Novel adalah suatu cerita yang memiliki alur panjang yang bersifat imajinatif, sehingga novel dapat dikatakan sebagaisuatu karya fiksi berupa prosa yang menceritakan tokoh-tokoh dalam suatu alur yang panjang (Tarigan, 2011). Novel memiliki alur yang cukup panjang dan memiliki kompleksitas tertentu yang berisikan rangkaian kejadian terkait yang melibatkan sekelompok orang dalam lingkungan tertentu.

2.3 Collaborative Filtering

Melakukan rekomendasi dengan menggunakan *Collaborative Filtering* dapat memperoleh kemiripan dari *user* atau *item* melalui analisis *rating* pengguna pada *item*. Selanjutnya dilakukan prediksi rating pada *item* yang tidak diberikan rating oleh *user*. Rating tertinggi digunakan sebagai rekomendasi untuk pengguna (Zhang & Song, 2009). Teknik *Collaborative Filtering* merupakan teknik yang paling populer dan banyak digunakan dalam sistem rekomendasi. Terdapat tiga jenis *Collaborative Filtering* seperti *memory based*, *model based*, dan *hybrid model*. *Memory Based Collaborative Filtering* dianggap sebagai aspek yang penting dalam sistem rekomendasi karena penggunaan yang mudah, memberikan fasilitas data baru dengan mudah, konten yang bebas dari elemen yang direkomendasikan untuk memperluas ruang lingkup untuk menghasilkan rating yang bagus. *Memory Based Collaborative Filtering* bekerja berdasarkan data yang telah ada dan melakukan prediksi pada rating. *Memory Based Collaborative Filtering* bergantung pada pengguna yang memiliki minat yang sama. *Memory Based Collaborative Filtering* memiliki dua jenis pendekatan yaitu *item based* dan *user based* (Mustafa, et al., 2017).

2.3.1 Item Based Collaborative Filtering

Item based collaborative filtering hampir mirip dengan metode *model based* dalam pembuatan rekomendasi. Terdapat dua fase dalam *item based collaborative filtering*. Pertama yaitu menemukan persamaan antara *item* yang dihitung dengan menggunakan satu jumlah *similarity measure*. Selanjutnya menghitung nilai *similarity* dengan menggunakan prediksi pada rating untuk *item* yang tidak diketahui nilai ratingnya (Mustafa, et al., 2017). Untuk menghitung kesamaan antar *item* dengan menggunakan *cosine similarty* dapat menggunakan persamaan 2.1

$$sim(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_u)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_u)^2}} \quad (2.1)$$

Keterangan:

$sim(i, j)$: nilai kemiripan

$u \in U$: *user* u yang melakukan *rating* pada genre i dan j

- $R_{u,i}$: rating yang diberikan *user u* pada genre *i*
- $R_{u,j}$: rating yang diberikan *user u* pada genre *j*
- \bar{R}_u : nilai rata-rata rating *user u*

Selanjutnya menghitung prediksi rating dengan mengukur kesamaan antara *item* target dengan *item* lain yang telah diberi rating oleh *user*. Untuk menghitung prediksi rating dapat menggunakan persamaan 2.2

$$P(u, j) = \frac{\sum_{i \in j} (R_{u,i} \times S_{i,j})}{\sum_{i \in j} |S_{i,j}|} \tag{2.2}$$

Keterangan:

- $P(u, j)$: nilai prediksi yang diberikan *user u* pada genre *j*
- $i \in j$: genre *i* yang mirip dengan genre *j*
- $R_{u,i}$: rating yang diberikan *user u* pada genre *i*
- $S_{i,j}$: nilai kemiripan antara genre *i* dengan genre *j*

2.4 Text Similarity

Text similarity merupakan salah satu metode untuk membuat sistem penilaian otomatis. *Text similarity* memiliki beberapa tahapan seperti *text preprocessing* dan perhitungan TF-IDF, serta penilaian menggunakan cosine similarity. Terdapat lima tahapan proses dalam melakukan *text preprocessing* yaitu *tokenization*, *cleaning*, *case folding*, *filtering*, dan *stemming*. Selanjutnya dilakukan proses perhitungan TF-IDF dan *cosine similarity* untuk mengetahui nilai kedekatan antara dua dokumen (Sulistyo, et al., 2015).

2.4.1 Tokenization

Tahap *tokenization* atau tokenisasi merupakan proses memecah kalimat menjadi beberapa bagian. Dalam tokenisasi, kata-kata pada tiap dokumen akan dihilangkan tanda baca serta simbol yang terdapat pada dokumen tersebut (Utami & Wahono, 2015). Tabel 2.2 merupakan contoh tokenisasi teks dokumen Bahasa Inggris.

Tabel 2. 2 Contoh *tokenization* teks dokumen Bahasa Inggris

Input	Output
In a future so real and near it might be now, something happens when women go to sleep; they become shrouded in a cocoon - like gauze.	"In"; "a"; "future"; "so"; "real"; "and"; "near"; "it"; "might"; "be"; "now,"; "something"; "happens"; "when"; "women"; "go"; "to"; "sleep;"; "they"; "become"; "shrouded"; "in"; "a"; "cocoon"; "-"; "like"; "gauze.";



2.4.2 Cleaning

Cleaning merupakan proses penghapusan tanda baca, angka, dan kata yang hanya mengandung satu atau dua huruf. Tabel 2.3 merupakan contoh *cleaning* teks dokumen Bahasa Inggris.

Tabel 2. 3 Contoh *cleaning* teks dokumen Bahasa Inggris

Input	Output
"In"; "a"; "future"; "so"; "real"; "and"; "near"; "it"; "might"; "be"; "now,"; "something"; "happens"; "when"; "women"; "go"; "to"; "sleep;"; "they"; "become"; "shrouded"; "in"; "a"; "cocoon"; "-"; "like"; "gauze.";	"future"; "real"; "and"; "near"; "might"; "now"; "something"; "happens"; "when"; "women"; "sleep"; "they"; "become"; "shrouded"; "cocoon"; "like"; "gauze";

2.4.3 Case Folding

Case folding merupakan suatu proses perubahan semua huruf dalam dokumen *lower case* (Wahyuni, et al., 2017). Tabel 2.4 merupakan contoh *case folding* pada dokumen Bahasa Inggris.

Tabel 2. 4 Contoh *case folding* teks dokumen Bahasa Inggris

Input	Output
"future"; "real"; "and"; "near"; "might"; "now"; "something"; "happens"; "when"; "women"; "sleep"; "they"; "become"; "shrouded"; "cocoon"; "like"; "gauze";	"future"; "real"; "and"; "near"; "might"; "now"; "something"; "happens"; "when"; "women"; "sleep"; "they"; "become"; "shrouded"; "cocoon"; "like"; "gauze";

2.4.4 Filtering

Filtering merupakan proses penghilangan kata-kata yang dianggap tidak penting atau jarang dicari atau digunakan sebagai *keywords* pada proses pencarian. *Filtering* digunakan untuk mengurangi waktu perhitungan frekuensi kata pada dokumen (Kurniawan, et al., 2014). Dalam proses *filtering* kata-kata yang tidak relevan atau tidak penting akan dihapus, seperti kata "a", "every", "please", dan sebagainya. Tabel 2.5 merupakan contoh *filtering* pada dokumen teks Bahasa Inggris.

Tabel 2. 5 Contoh *filtering* pada dokumen teks Bahasa Inggris

Input	Output
"future"; "real"; "and"; "near"; "might"; "now"; "something"; "happens"; "when"; "women";	"future"; "real"; "near"; "happens"; "women"; "sleep"; "shrouded";

“sleep”; “they”; “become”; “shrouded”; “cocoon”; “like”; “gauze”;	“cocoon”; “gauze”; “awakened”; “gauze”;
--	--

2.4.5 Stemming

Pengertian mengenai *stemming* yaitu suatu proses perubahan kata berimbuhan atau penghapusan imbuhan pada suatu kata berimbuhan sehingga menjadi kata dasar (Utami & Wahono, 2015). Tabel 2.6 merupakan contoh *stemming* pada teks dokumen Bahasa Inggris.

Tabel 2. 6 Contoh *stemming* pada teks dokumen Bahasa Inggris

Input	Output
“future”; “real”; “near”; “happens”; “women”; “sleep”; “shrouded”; “cocoon”; “gauze”; “awakened”; “gauze”;	“future”; “real”; “near”; “happen”; “women”; “sleep”; “shroud”; “cocoon”; “gauze”; “awaken”; “gauze”;

2.4.6 TF-IDF

TF-IDF merupakan proses penggabungan dua konsep untuk perhitungan bobot yaitu dengan cara mengalikan frekuensi kemunculan term dalam sebuah dokumen dengan *inverse* frekuensi dokumen yang mengandung term tersebut. Tingkat seringnya kemunculan term dalam dokumen menunjukkan tingkat kepentingan term tersebut dalam dokumen, sedangkan tingkat umumnya suatu term tersebut dilihat dari tingkat kemunculan dokumen yang mengandung term. Bobot term bernilai tinggi apabila frekuensi term tersebut banyak (Wahyuni, et al., 2017). Perhitungan TF-IDF dijabarkan pada persamaan 2.3.

$$TF - IDF = \log(1 + t_{f_{t,d}}) \times \log_{10}\left(\frac{N}{df_t}\right) \quad (2.3)$$

Keterangan:

TF-IDF : nilai bobot *term t* dokumen *d*

$t_{f_{t,d}}$: frekuensi kemunculan *term t* pada suatu dokumen *d*

N : jumlah dokumen

df_t : frekuensi kemunculan dokumen yang mengandung *term t*

2.4.7 Cosine Similarity

Cosine similarity adalah suatu perhitungan kesamaan antara dua vektor dengan mencari kosinus sudut antara kedua vektor dimana nilai terkecil yaitu nol dan nilai terbesar adalah satu (Kesuma & Pribadi, 2016). *Cosine similarity* digunakan dalam melakukan perhitungan kesamaan dari dokumen (Wahyuni, et

al., 2017). Rumus dalam menghitung *cosine similarity* dijabarkan pada persamaan 2.4

$$\text{Cos } \alpha = \frac{A \cdot B}{|A||B|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (2.4)$$

Keterangan:

A : vektor A

B : vektor B

$A \cdot B$: perkalian antara vektor A dengan vektor B

$|A|$: panjang vektor A

$|B|$: panjang vektor B

$|A||B|$: perkalian antara $|A|$ dan $|B|$

2.5 Accuracy

Akurasi yang dihasilkan dari prediksi dapat dihitung setelah jumlah data yang telah diprediksi benar dan salah telah diketahui (Devasena, 2013). Untuk menghitung nilai akurasi digunakan persamaan 2.8.

$$\text{Accuracy} = \frac{\text{jumlah data yang diprediksi secara benar}}{\text{jumlah prediksi yang dilakukan}} \times 100\% \quad (2.8)$$

2.6 Mean Absolute Error (MAE)

Mean Absolute Error (MAE) digunakan untuk mengevaluasi kualitas sistem rekomendasi. MAE merupakan ukuran penyimpangan rekomendasi nilai yang ditentukan pengguna yang sebenarnya. Semakin rendah nilai MAE maka semakin akurat rekomendasi yang dihasilkan dari hasil prediksi rating oleh pengguna. Untuk menghitung MAE dilakukan penjumlahan kesalahan-kesalahan absolut dari pasangan prediksi rating yang sesuai dan kemudian menghitung rata-rata (Zhang & Song, 2009). Untuk menghitung MAE digunakan persamaan 2.9.

$$\text{MAE} = \frac{\sum_{i=1}^N |p_i - q_i|}{N} \quad (2.9)$$

Keterangan:

p_i : nilai rating yang diprediksi

q_i : nilai rating sebenarnya

N : jumlah rating pada semua pengguna

BAB 3 METODOLOGI PENELITIAN

3.1 Tipe Penelitian

Jenis penelitian yang digunakan adalah penelitian non-implimentatif di mana penelitian ini berfokus pada proses investigasi terhadap keadaan atau situasi tertentu serta analisis tentang hubungan antar kejadian yang sedang dikaji dan selanjutnya akan menghasilkan produk berupa suatu analisis yang bersifat ilmiah. Metode yang digunakan dalam menghasilkan produk utama dapat berupa obeservasi, kuisisioner, survei, eksperimen, dan sebagainya. Jika ditinjau dan dilihat dari aktivitas penelitiannya, pendekatan yang diguakan pada penelitian ini merupakan pendekatan non-implimentatif analitik. Pendekatan jenis ini bertujuan untuk menjelaskan hubungan antar elemen pada objek penelitian yang sedang diteliti dengan situasi tertentu dan menghasilkan produk berupa suatu analisis.

3.2 Strategi Penelitian

Penelitian ini dapat dikategorikan sebagai penelitian yang menggunakan metode eksperimen. Penelitian eksperimen merupakan penelitian yang menggunakan investigasi hubungan sebab dan akibat. Uji coba dalam melakukan penelitian ini dikontrol oleh peneliti dengan mengembangkan dan melakukan evaluasi. Metode eksperimen yang digunakan dalam penelitian ini adalah *Genre Based Collaborative Filtering* dan *Text Similarity*.

3.3 Partisipan Penelitian

Kriteria partisipan yang ikut terlibat dalam penelitian ini yaitu seseorang yang gemar membaca berbagai macam novel dan melakukan *review* atau ulasan novel yang telah dibaca pada situs *goodreads.com*. Alasan memilih partisipan penelitian ini agar data yang dihasilkan berupa data yang valid.

3.4 Lokasi Penelitian

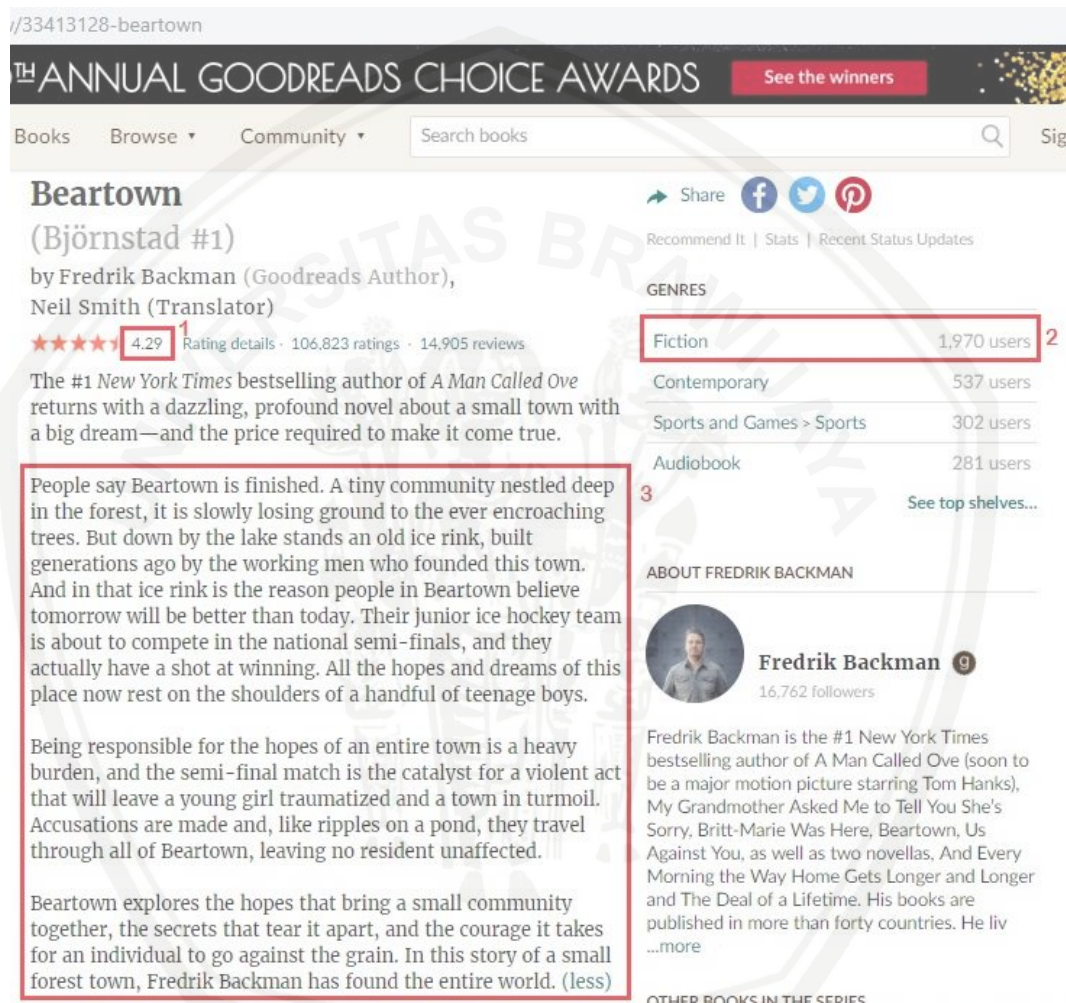
Penelitian ini dilakukan di Laboratorium Komputasi Cerdas Fakultas Ilmu Komputer Universitas Brawijaya. Alasan pemilihan lokasi penelitian dilakukan yaitu untuk melakukan pengukuran prediksi rating novel baru berbahasa Inggris pada situs *goodreads.com* dengan menggunakan *genre based collaborative filtering* dan *text similarity*.

3.5 Teknik Pengumpulan Data

Pengumpulan data bertujuan untuk mengumpulkan data dan informasi yang kemudian akan diolah dan digunakan untuk menjawab rumusan masalah pada penelitian. Dataset yang digunakan berupa data primer yang diambil dari situs *goodreads.com*. Dataset yang diambil berupa data latih dan data uji. Proses pengumpulan data yaitu melakukan observasi terhadap beberapa *user* yang melakukan *review* dan memberikan rating pada buku yang sama. Terdapat empat

fitur yang digunakan yaitu *user*, genre, rating, dan sinopsis. Berikut ini merupakan langkah-langkah pengambilan data:

1. Pemilihan sinopsis dilakukan secara acak. Kemudian mengambil rating yang terdapat pada bawah *author* atau apabila pada gambar 3.1 terdapat pada nomor 1. Kemudian untuk genrenya diambil dari “genres” yang berada di sebelah kanan dan dipilih yang paling atas yang mana genre tersebut merupakan genre yang paling banyak dipilih oleh *user* atau apabila pada gambar 3.1 terdapat pada nomor 2. Selain itu mengambil isi sinopsisnya atau apabila pada gambar 3.1 terdapat pada nomor 3.



Gambar 3. 1 Pengambilan data berupa rating, genre, serta sinopsis

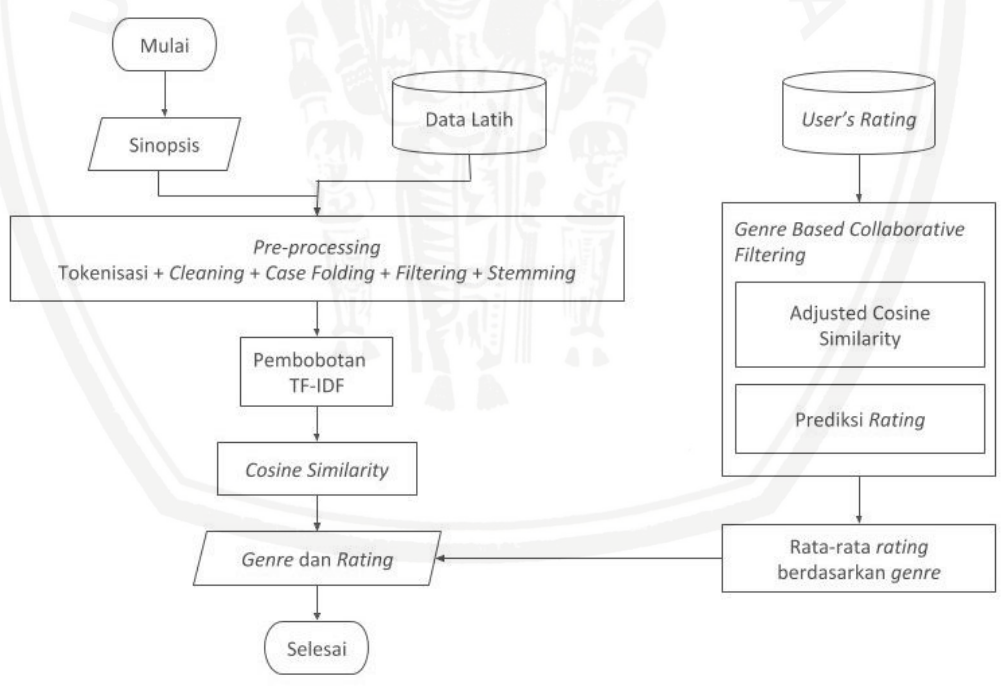
2. Selanjutnya mengambil rating yang dimasukkan oleh *user goodread.com*. User yang dipilih yaitu user yang memberikan rating pada banyak sinopsis. Kemudian rating yang diambil yaitu jumlah bintang yang diberikan oleh *user goodreads.com*.



Gambar 3. 2 Pengambilan rating yang diberikan oleh user

3.6 Implementasi Algoritme

Dalam pembangunan suatu sistem diperlukan penjabaran proses-proses secara detail yang kemudian disebut sebagai implementasi algoritme. Sistem ini bertujuan untuk memberikan prediksi rating pada novel berdasarkan sinopsisnya. Gambaran umum mengenai perancangan algoritme ditunjukkan pada Gambar 3.1



Gambar 3. 3 Perancangan Algoritme

3.7 Teknik Analisis Data

Proses pengujian dilakukan setelah penyusunan dan pembuatan kode program dari kerangka rancangan algoritme. Proses pengujian ini dilakukan dengan tujuan untuk mengetahui kesesuaian sistem yang dibuat dengan kebutuhan dan perancangan yang dibuat. Proses pengujian dilakukan dengan membandingkan hasil peratingan novel yang terdapat pada situs *goodreads.com* dengan yang dilakukan oleh sistem.

Setelah proses pengujian selesai dilaksanakan, maka selanjutnya dilakukan analisis terhadap hasil dari pengujian tersebut. Tahap analisis dilakukan dengan melihat nilai akurasi yang dihasilkan oleh *Text Similarity* dan *Genre Based Collaborative Filtering* dari skenario pengujian untuk menilai prediksi yang dihasilkan oleh metode *Genre Based Collaborative Filtering*.

3.8 Kesimpulan dan Saran

Setelah seluruh tahapan penelitian selesai dilakukan, langkah terakhir yang perlu dilakukan yaitu pengambilan kesimpulan dan pemberian saran. Pembuatan kesimpulan didasarkan pada hasil pengujian.

Kesimpulan menjawab pertanyaan-pertanyaan yang terdapat pada rumusan masalah yang telah ditentukan meliputi prediksi rating pada novel baru berdasarkan sinopsis menggunakan *Genre Based Collaborative Filtering* dan *Text Similarity* dan menghitung tingkat akurasi yang dihasilkan *Genre Based Collaborative Filtering* dan *Text Similarity* pada sinopsis novel untuk melakukan prediksi rating pada novel baru.

Saran diperoleh dari hasil analisis yang dilakukan pada pengujian. Penulisan saran bertujuan untuk memberikan rekomendasi perbaikan terhadap kesalahan-kesalahan yang ada serta sebagai pengembangan untuk penelitian selanjutnya.

BAB 4 PERANCANGAN

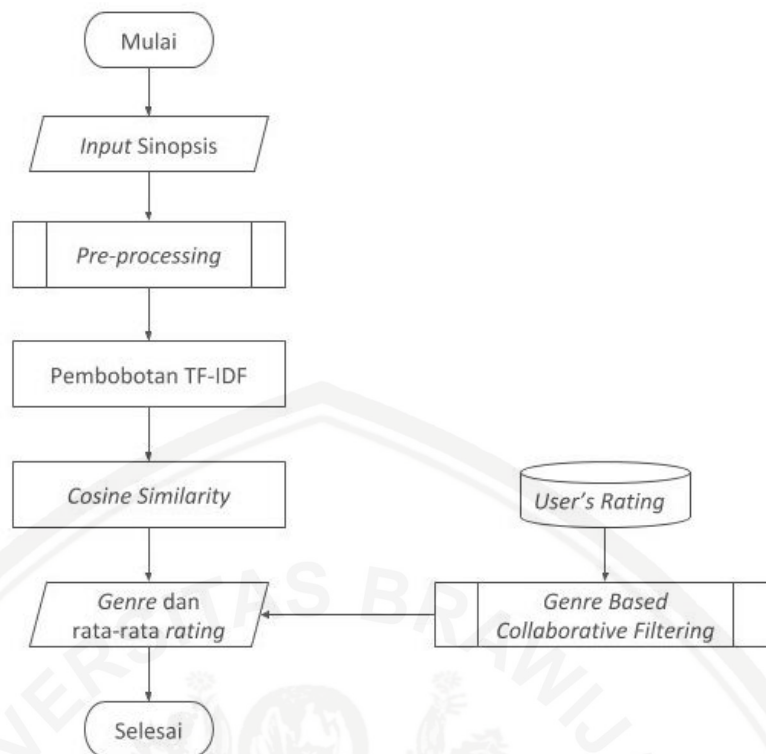
4.1 Deskripsi permasalahan

Novel merupakan salah satu produk berupa cerita yang memiliki alur panjang yang bersifat imajinatif, sehingga novel dapat dikatakan sebagai suatu karya fiksi berupa prosa yang menceritakan tokoh-tokoh dalam suatu alur yang panjang. Banyaknya novel yang terbit tiap tahunnya membuat calon pembaca bingung dalam memilih novel baru yang hendak dibaca. Salah satu acuan dalam menentukan kualitas novel diperlukan rating suatu program atau produk berdasarkan respon masyarakat terhadap program atau produk. Namun untuk novel baru tentunya masih sedikit atau bahkan belum ada yang melakukan review ataupun memberikan rating sebagai respon oleh masyarakat, sehingga diperlukan suatu prediksi untuk mengetahui rating novel baru tersebut.

Dalam melakukan prediksi terdapat beberapa metode yang digunakan, salah satunya yaitu *collaborative filtering*. Metode *collaborative filtering* dapat digunakan untuk melakukan prediksi rating yang diberikan oleh *user*. Namun dalam novel baru tentunya masih sedikit atau belum ada *user* yang memberikan rating, sehingga digunakan sinopsis pada novel untuk mengetahui kesamaan teks antara novel baru dengan novel lain. Dari kesamaan teks dapat diklasifikasikan *genre* dari novel baru tersebut dan prediksi rating yang diberikan oleh *user* didapat dari rata-rata rating oleh *user*.

4.2 Deskripsi umum sistem

Sistem dalam penelitian ini bertujuan untuk menyelesaikan permasalahan prediksi mengenai novel baru berbahasa Inggris menggunakan *genre based collaborative filtering* dan *text similarity*. *Genre based collaborative filtering* digunakan untuk melakukan prediksi rating yang telah diberikan oleh *user* pada novel data latih. Terdapat dua fase dalam *genre based collaborative filtering* dalam melakukan prediksi rating. Pertama yaitu menemukan persamaan antara *genre* yang dihitung dengan menggunakan *similarity measure*. Selanjutnya menghitung nilai prediksi pada rating untuk *genre* yang tidak diketahui nilai ratingnya. Dokumen yang berupa sinopsis akan diprediksi genrenya. diawali dengan melakukan proses *pre-processing*. Terdapat beberapa tahap dalam proses *pre-processing* yaitu *tokenization*, *cleaning*, *case folding*, *filtering*, hingga *stemming*. Setelah proses *pre-processing* kemudian melakukan pembobotan kata dengan menggunakan TF-IDF. Setelah dilakukan pembobotan kata kemudian dilakukan perhitungan menggunakan *cosine similarity* untuk melakukan klasifikasi dokumen berdasarkan genrenya. Dari perhitungan dengan menggunakan *cosine similarity* dapat diketahui kesamaan sinopsis dari masukan pada sistem dengan data latih. Kemudian dari kesamaan tersebut diketahui *genre* yang mendekati sinopsis dari masukan. Kemudian prediksi rating untuk novel baru didapat dari rata-rata rating pada *genre* yang sama. Deskripsi umum sistem digambarkan pada Gambar 4.1



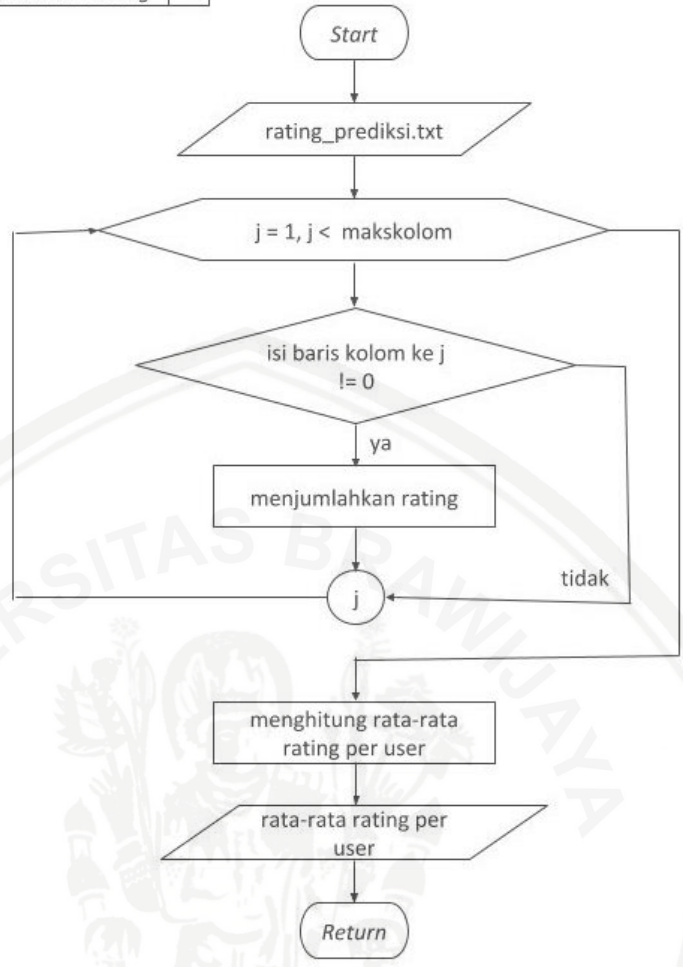
Gambar 4. 1 Deskripsi umum sistem

Pada Gambar 4.1 menunjukkan bahwa pada diagram alir sistem memiliki beberapa tahapan untuk mengetahui prediksi rating pada novel baru. Proses pertama yang dilakukan yaitu melakukan prediksi rating novel yang belum diberikan rating oleh user pada data latih. Selanjutnya dilakukan proses *pre-processing* pada dokumen yang berupa sinopsis. Kemudian dari hasil *pre-processing*, akan dilakukan pembobotan kata dengan menggunakan TF-IDF. Selanjutnya dilakukan perhitungan dengan menggunakan *cosine similarity* untuk menentukan genre dari novel baru tersebut. Setelah diketahui genre dari novel baru tersebut maka akan dihitung rata-rata rating yang diberikan oleh user pada genre yang sama, yang selanjutnya rata-rata rating tersebut dijadikan sebagai prediksi rating pada novel baru.

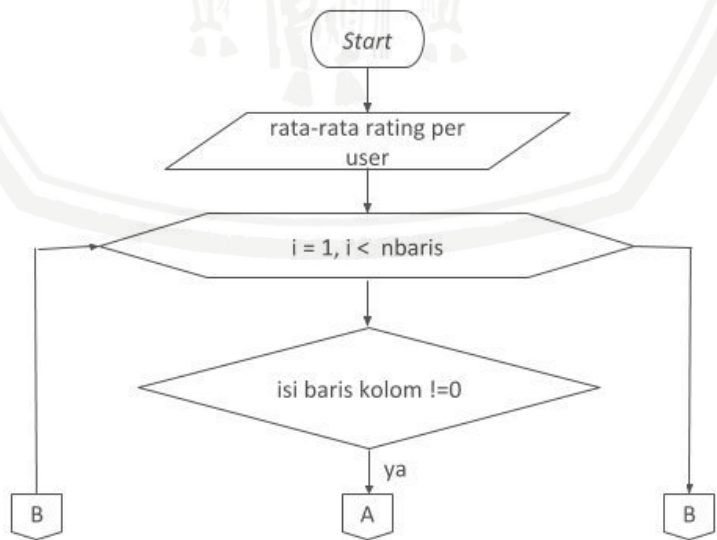
4.3 Genre Based Collaborative Filtering

Genre based collaborative filtering merupakan salah satu metode dalam memberikan rekomendasi berdasarkan kemiripan antar genre. *Genre based collaborative filtering* melakukan analisis terhadap matriks *user-genre* untuk mengetahui hubungan antar *genre* dan *user*. Penjelasan alir proses *genre based collaborative filtering* ditunjukkan pada gambar 4.2

Genre Based Collaborative Filtering

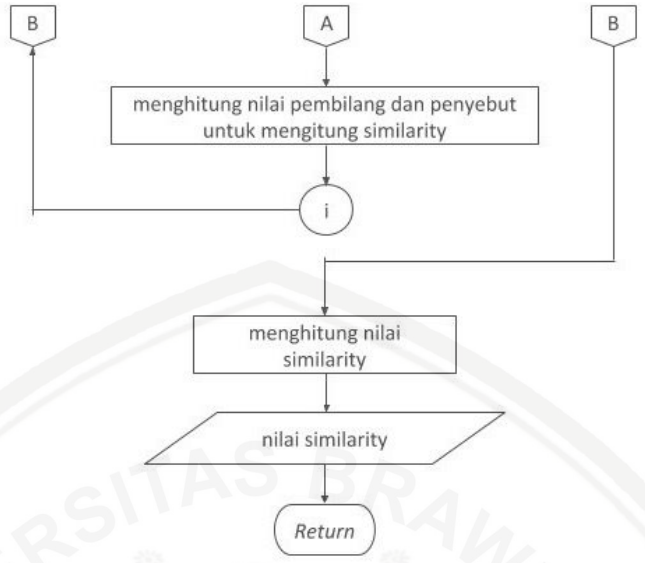


Genre Based Collaborative Filtering

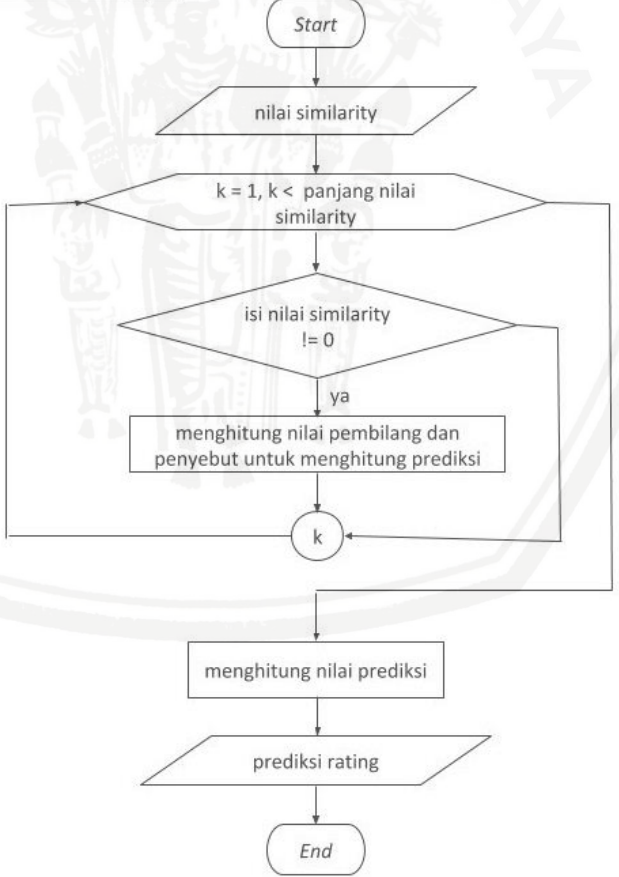




Genre Based Collaborative Filtering



Genre Based Collaborative Filtering

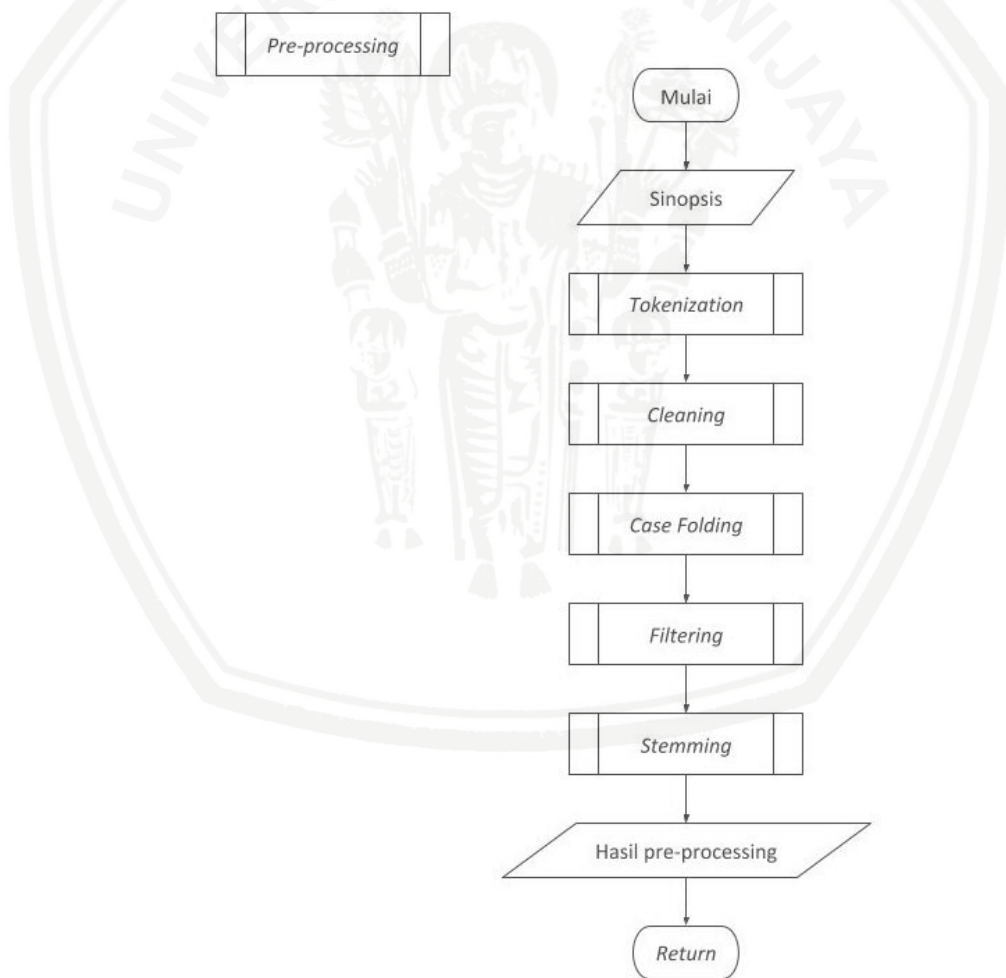


Gambar 4. 2 Alir proses genre based collaborative filtering

Gambar 4.2 menjelaskan alur proses *genre based collaborative filtering*. Dimulai dari masukan berupa file `rating_prediksi.txt` yang berisi rating dari *user* yang telah ditentukan. Kemudian dilakukan perhitungan rata-rata rating per *user*. hasil dari perhitungan tersebut akan digunakan untuk menghitung nilai pembilang dan penyebut yang selanjutnya digunakan dalam menghitung nilai *similarity* dengan *adjusted cosine similarity*. Langkah selanjutnya yaitu menghitung prediksi rating yang masih kosong. Untuk menghitung prediksi dilakukan pembagian antara nilai *similarity* antar genre yang dikalikan dengan rating yang diberikan *user* pada genre tertentu dengan nilai *similarity* antar genre. Hasil dari perhitungan prediksi tersebut kemudian dihitung rata-rata tiap genrenya. Hasil dari rata-rata tersebut yang kemudian dijadikan sebagai rating prediksi tiap genre.

4.4 Text Similarity

Salah satu tahap dalam text similarity yaitu text pre-processing yang bertujuan untuk mengubah dokumen yang tidak terstruktur menjadi beberapa bagian hingga menjadi dokumen yang terstruktur. Penjelasan alir proses teks pre-processing ditunjukkan Gambar 4.3

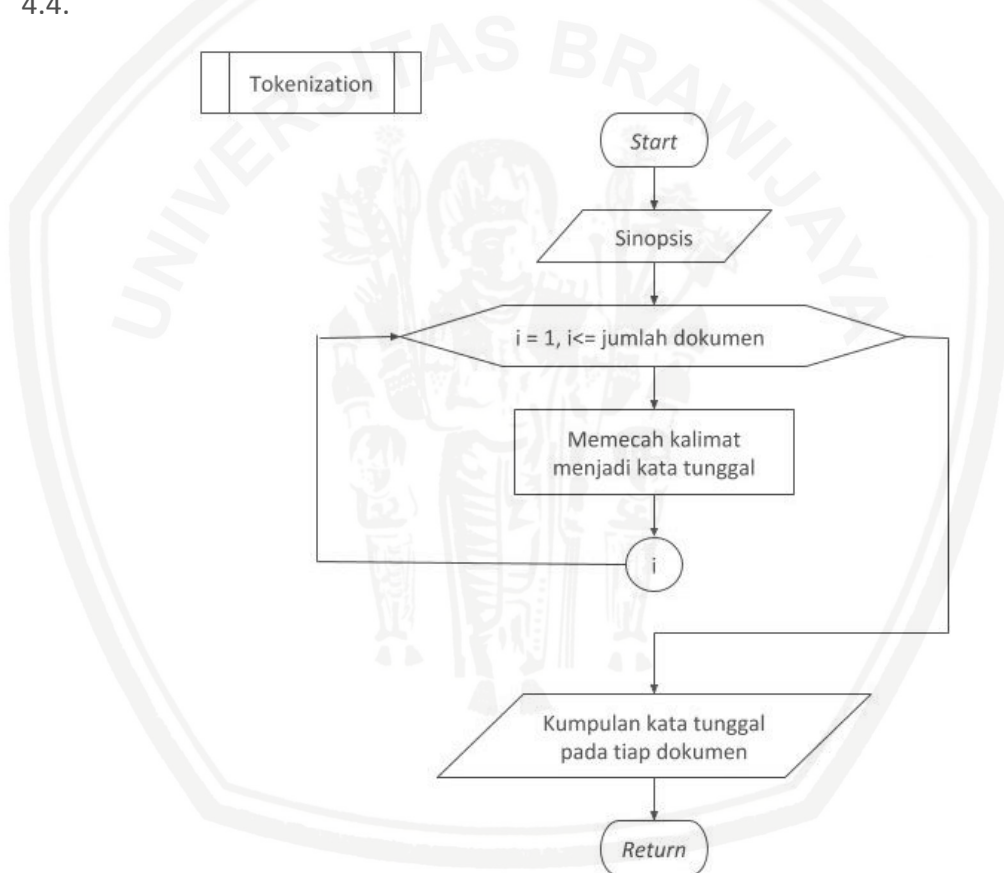


Gambar 4. 3 Diagram alir *pre-processing*

Gambar 4.3 menunjukkan terdapat beberapa tahapan dalam *pre-processing* yang harus dijalankan. Pada proses awal, data sebagai masukan berupa sinopsis novel. Selanjutnya akan diproses melalui proses *tokenization* atau proses pemecahan kalimat menjadi kata, *cleaning* atau penghapusan karakter, huruf dari bahasa lain, dan simbol, *case folding* pengubahan semua huruf menjadi huruf kecil, *filtering* atau penghapusan kata yang tidak diperlukan yang terdapat pada kamus *stopword*, serta *stemming* atau perubahan kata yang berimbunan menjadi kata dasar. Setelah dilakukan semua tahap *pre-processing* dihasilkan hasil dari *pre-processing* yang akan digunakan dalam tahap selanjutnya.

4.4.1 Tokenization

Tahap *tokenization* atau tokenisasi merupakan tahapan pertama dalam proses *pre-processing*. Dalam tahap *tokenization* dilakukan pemecahan kalimat menjadi kata. Penjelasan alir proses dari *tokenization* ditunjukkan pada Gambar 4.4.



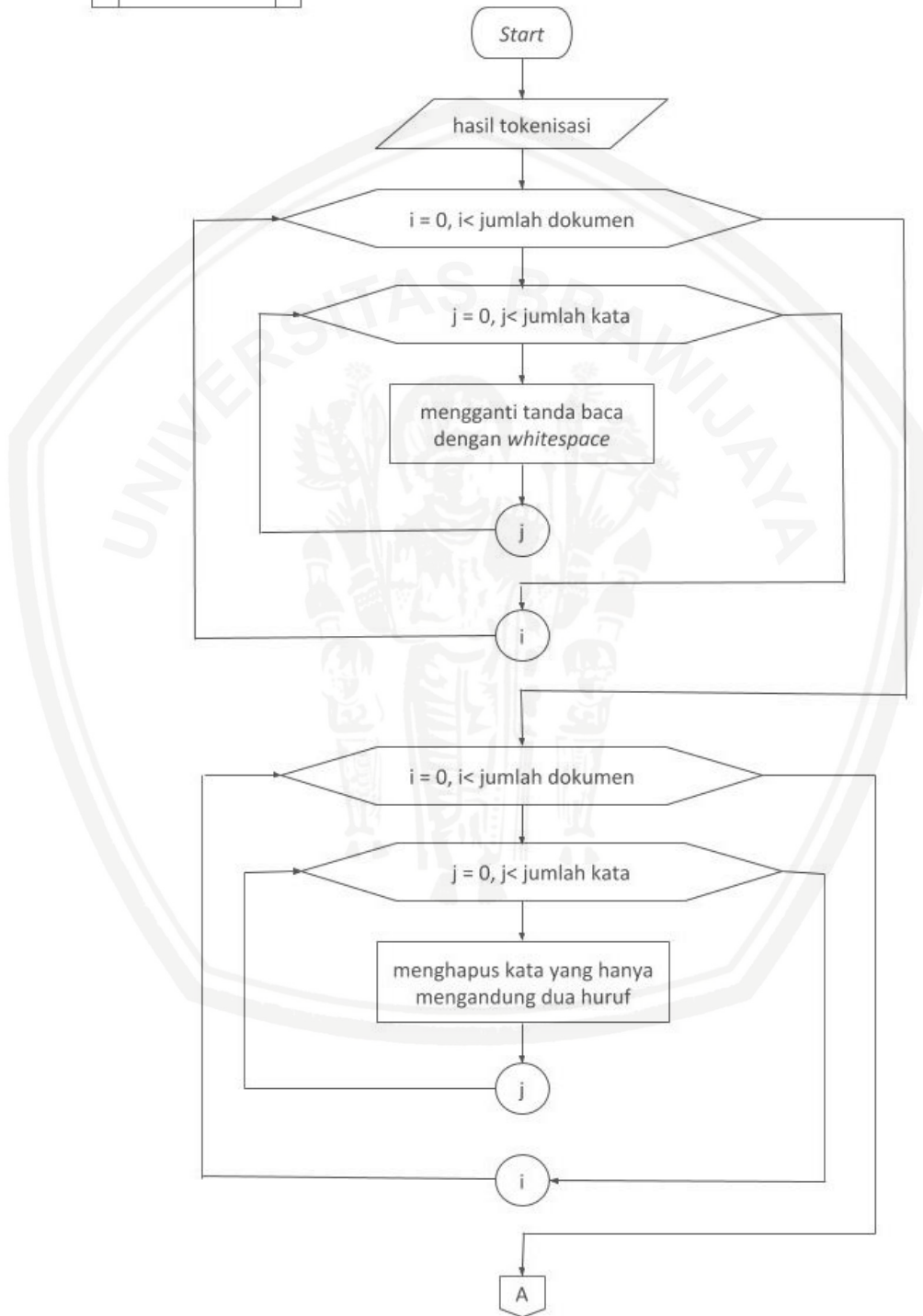
Gambar 4. 4 Diagram alir proses *tokenization*

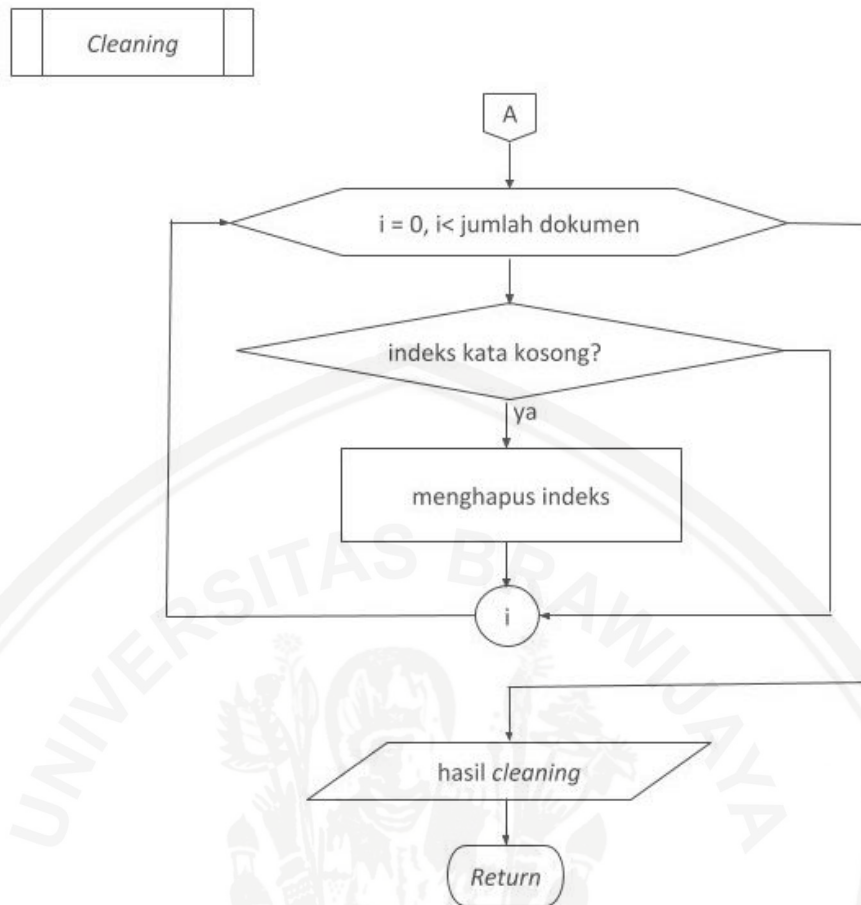
Gambar 4.4 menunjukkan dalam proses *pre-processing* dimulai dengan memasukkan berupa data sinopsis novel yang didapatkan dari goodreads.com yaitu data latih dan data uji. Selanjutnya data tersebut dipecah menjadi beberapa kata sehingga hasil keluaran berupa kumpulan kata pada setiap data.

4.4.2 Cleaning

Cleaning merupakan proses penghapusan tanda baca, angka, huruf dari bahasa asing, dan kata yang hanya mengandung satu atau dua huruf. Hasil dari proses *cleaning* akan digunakan untuk proses *pre-processing* selanjutnya. Penjelasan alir proses *cleaning* ditunjukkan pada Gambar 4.5

Cleaning





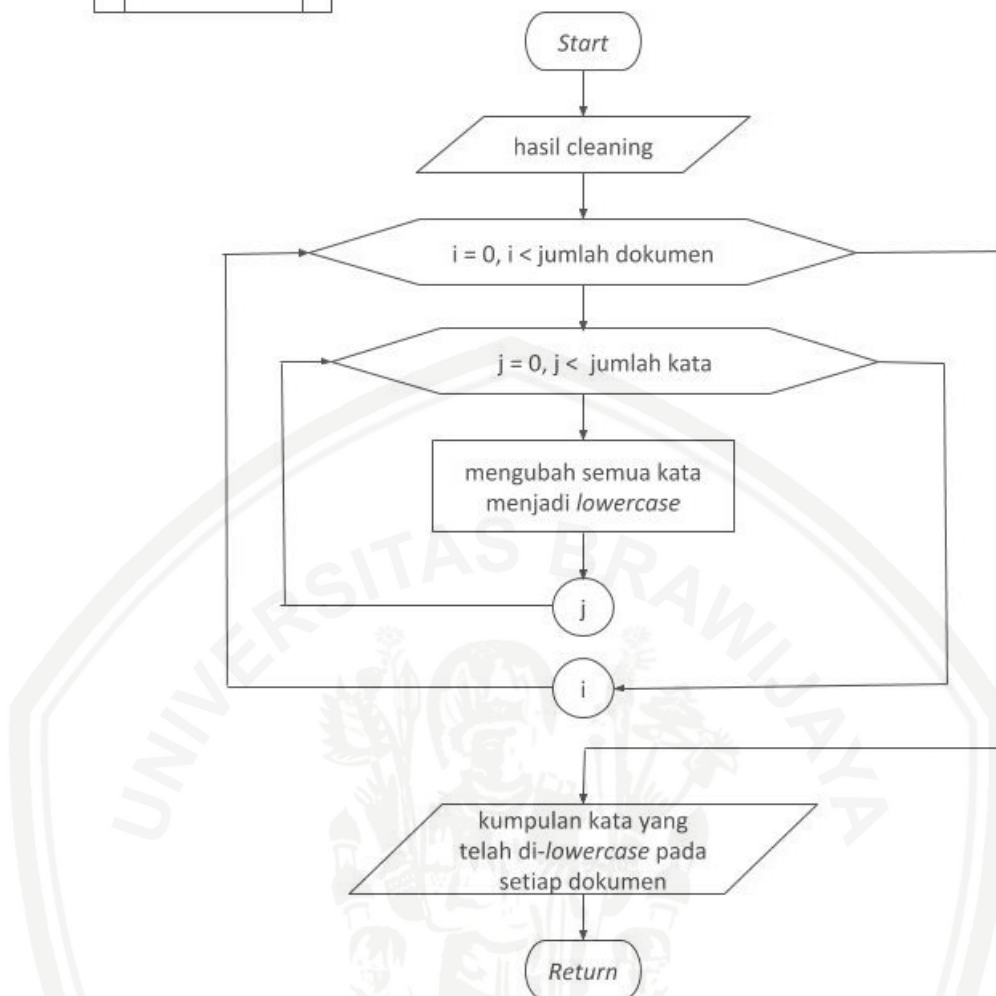
Gambar 4. 5 Diagram alir proses *cleaning*

Gambar 4.5 menunjukkan proses selanjutnya dalam *pre-processing* yaitu proses *cleaning*. Proses *cleaning* menggunakan *input* dari hasil tokenisasi pada proses sebelumnya. Kemudian *input* tersebut dilakukan proses penggantian tanda baca dengan *whitespace*. Selanjutnya penggantian huruf dari berbagai bahasa dan angka desimal dengan *whitespace*. Kemudian menghapus kata yang hanya mengandung satu atau dua huruf. Selanjutnya menghapus indeks yang kosong, sehingga keluaran yang dihasilkan merupakan hasil keseluruhan *cleaning*.

4.4.3 Case Folding

Tahap setelah dilakukan proses *cleaning* dalam *pre-processing* yaitu proses *case folding*. Dalam proses ini semua huruf dari hasil *cleaning* akan diubah menjadi huruf kecil atau yang biasa disebut sebagai *lower case*. Hasil dari *case folding* akan digunakan untuk proses selanjutnya. Penjelasan alir proses *cleaning* ditunjukkan pada Gambar 4.6

Case Folding



Gambar 4. 6 Diagram alir *case folding*

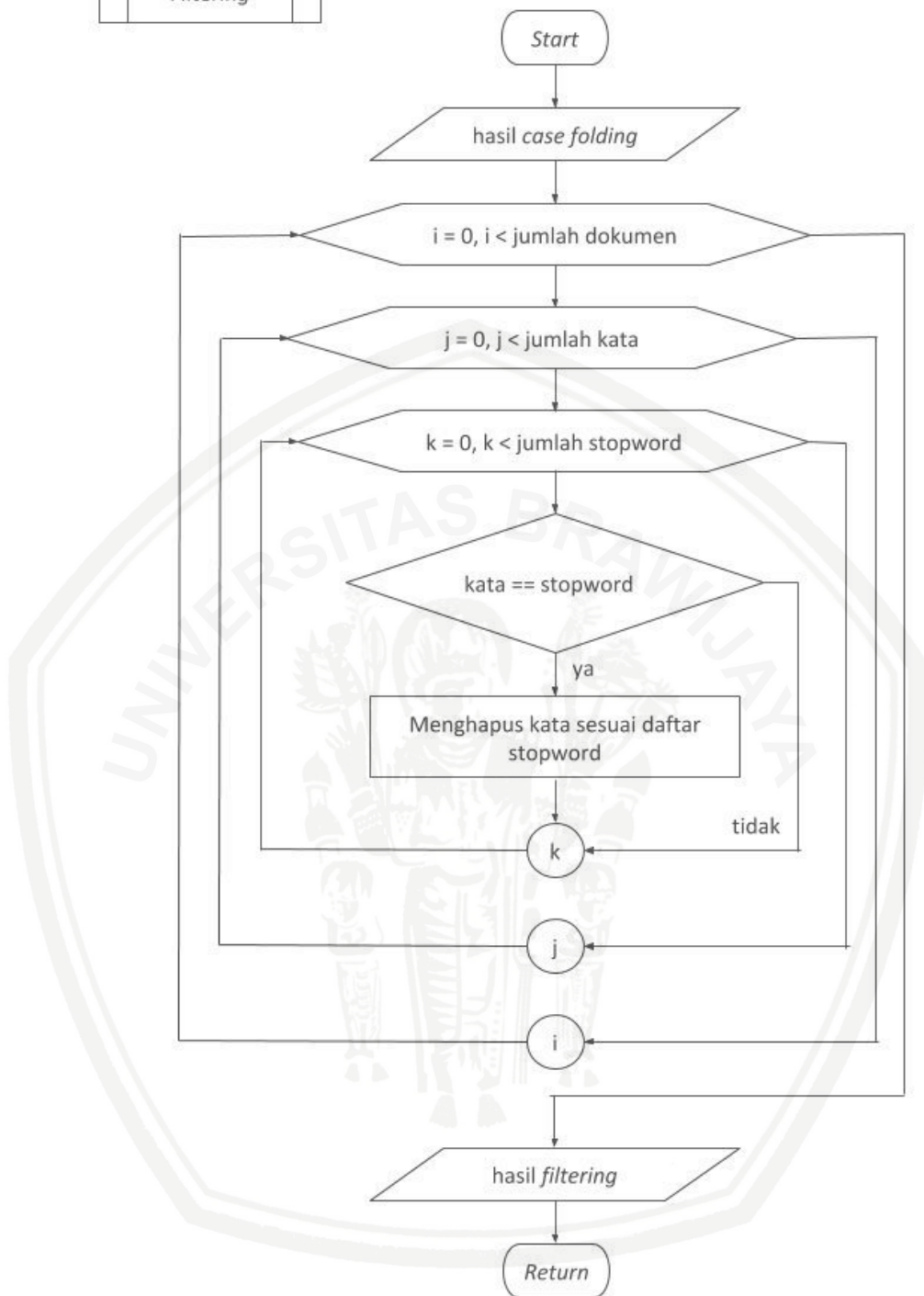
Gambar 4.6 menunjukkan proses *case folding*. Masukan dari proses *case folding* merupakan hasil dari proses sebelumnya, yaitu hasil dari proses *cleaning*. Selanjutnya hasil *cleaning* diatur untuk diubah menjadi huruf kecil atau *lower case* semua. Hasil dari *case folding* berupa kumpulan kata yang telah dilakukan proses *lower case* pada tiap dokumen. Hasil *case folding* akan digunakan untuk proses selanjutnya.

4.4.4 Filtering

Dalam pre-processing, tahap setelah *case folding* yaitu *filtering* . Dalam tahap *filtering* dilakukan proses untuk menghilangkan kata yang sama dengan kata yang terdapat pada kamus *stopword*. Hasil dari *filtering* ini akan digunakan untuk tahap *pre-processing* selanjutnya. Penjelasan alir dari proses *filtering* ditunjukkan pada gambar 4.7



Filtering



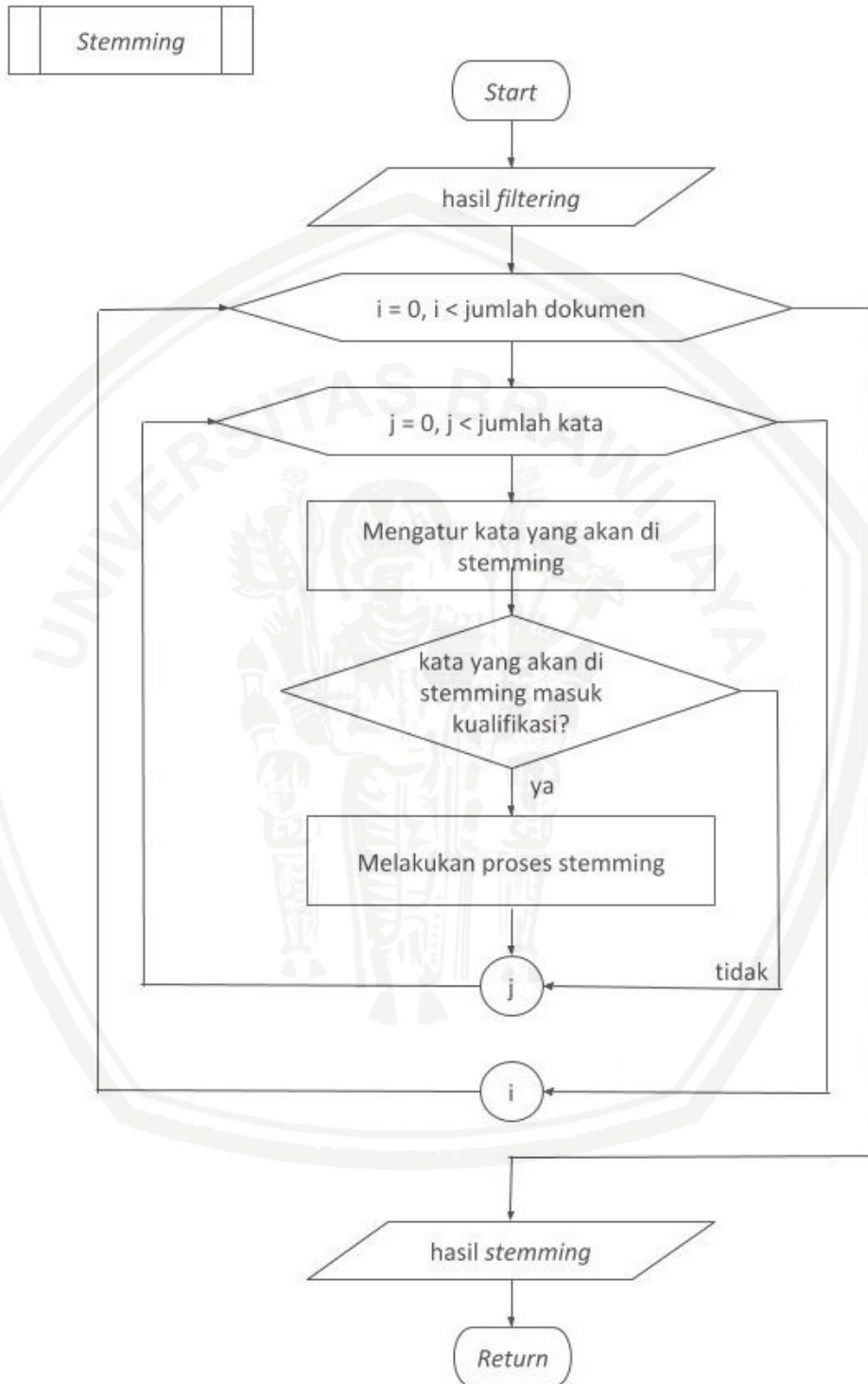
Gambar 4. 7 Diagram alir *filtering*

Gambar 4.7 menunjukkan alir dari proses *filtering*. Masukan dari proses ini merupakan hasil dari proses sebelumnya yaitu dari proses *case folding*. Hasil dari proses *case folding* kemudian akan dicek kesamaannya dengan kata-kata yang terdapat pada daftar *stopword*. Kata akan dihapus jika ditemukan kata yang sama antara kata dalam dokumen dengan kata dalam kamus *stopword*. Hasil dari proses *filtering* ini akan digunakan untuk proses selanjutnya.



4.4.5 Stemming

Tahap akhir dari *pre-processing* yaitu *stemming*. Dalam proses *stemming* kata berimbuhan akan dihapus dan hanya menyisakan kata dasar. Penjelasan alir proses *stemming* ditunjukkan pada Gambar 4.8

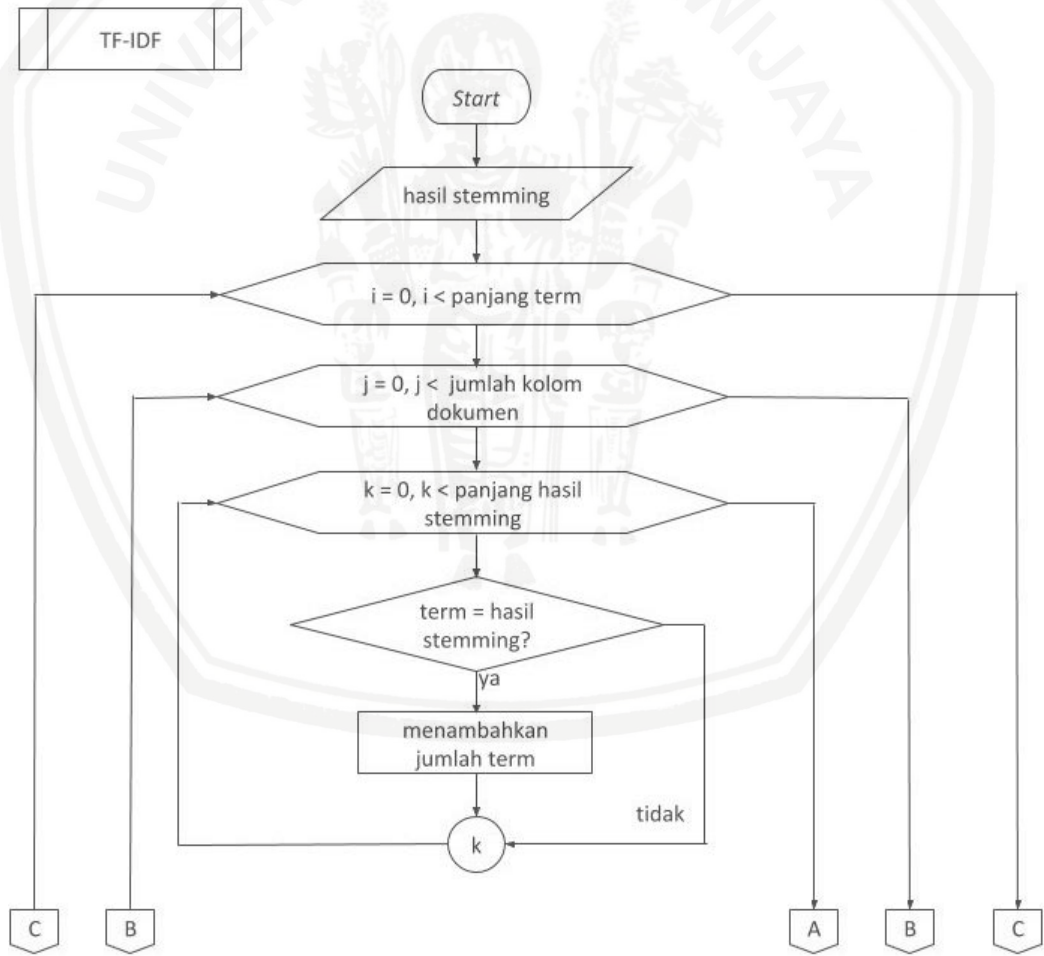


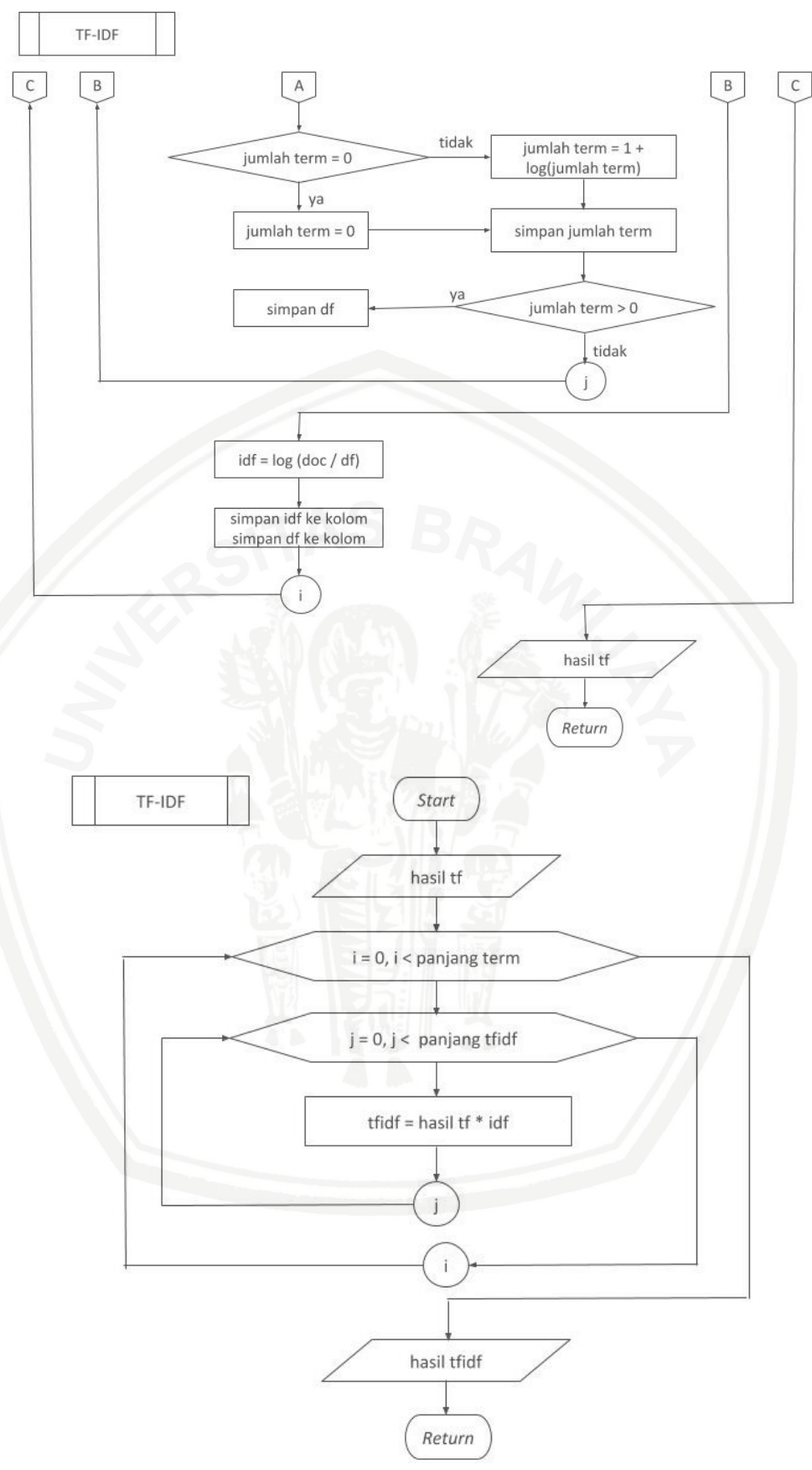
Gambar 4. 8 Diagram alir *stemming*

Gambar 4.8 menunjukkan alir dari *stemming*. Tujuan dari proses *stemming* ini yaitu untuk menghapus imbuhan pada kata yang berimbuhan sehingga hanya tersisa kata dasar. Proses *stemming* diawali dengan memasukkan kata yang berasal dari hasil proses *filtering* yang selanjutnya dilakukan proses *stemming* dengan menggunakan *library* dari *snowball.tartarus.org*. Hasil dari *stemming* akan digunakan untuk proses selanjutnya, yaitu pembobotan dengan menggunakan TF-IDF.

4.4.6 TF-IDF

TF-IDF merupakan salah satu pemebobotan term. TF-IDF dilakukan dengan cara mengalikan nilai TF atau frekuensi kemuculan term pada suatu dokumen dengan IDF atau nilai *inverse* dari frekuensi dokumen yang mengandung term tersebut. Tingkat frekuensi kemuculan term dalam suatu dokumen menunjukkan tingkat pentingnya term tersebut dalam dokumen, sedangkan tingkat umumnya suatu term tersebut dilihat dari tingkat kemunculan dokumen yang mengandung term. Bobot term bernilai tinggi apabila frekuensi term tersebut banyak. Penjelasan alir proses TF-IDF ditunjukkan pada Gambar 4.9



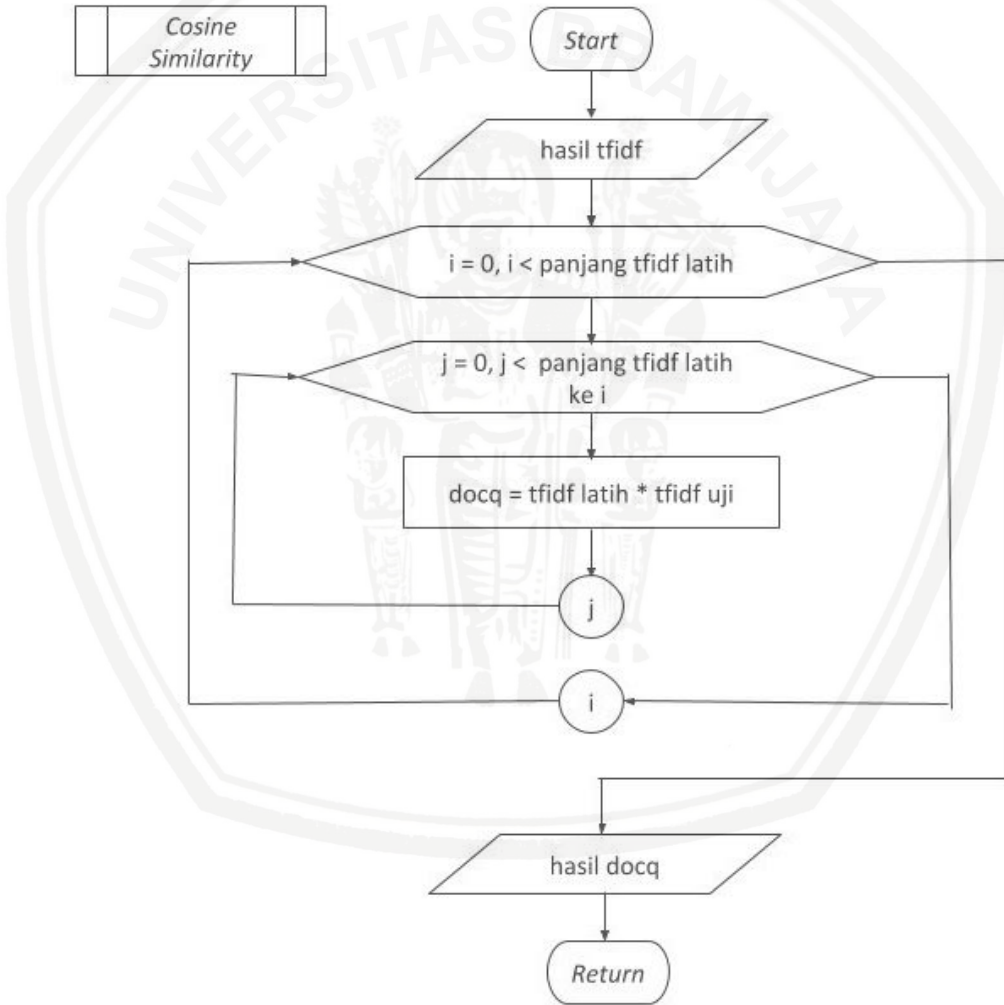


Gambar 4. 9 Diagram alir proses TF-IDF

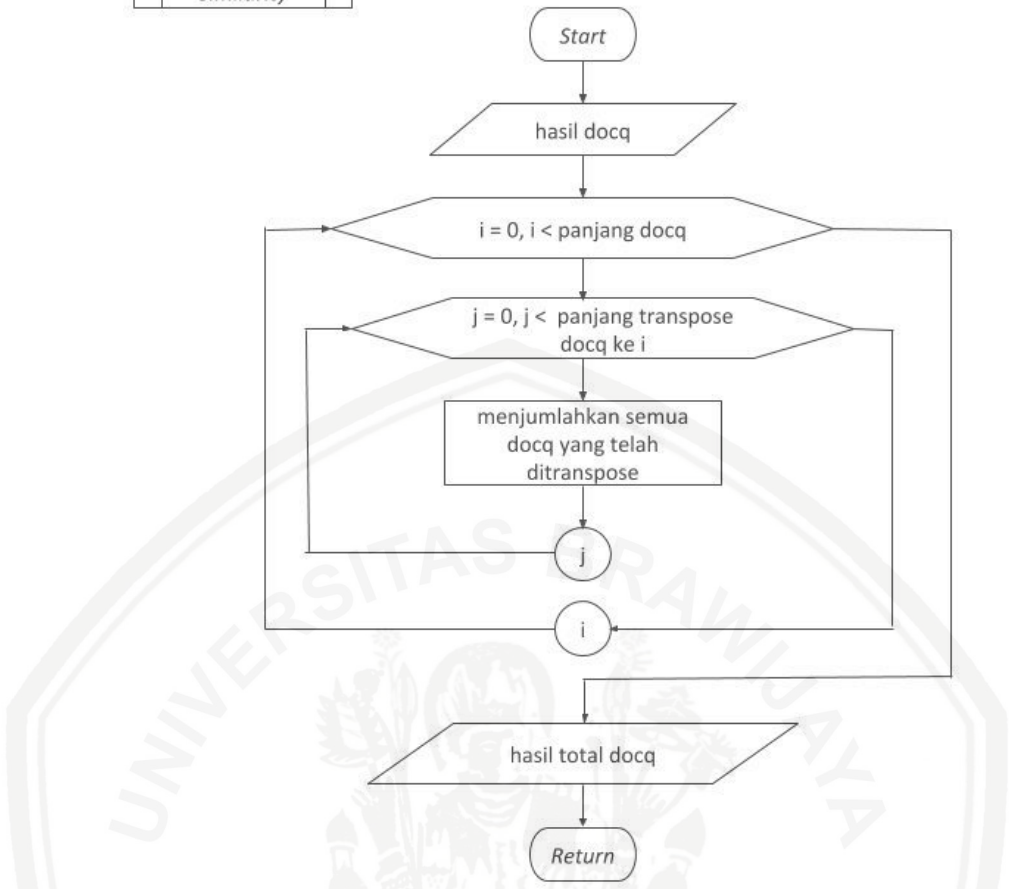
Gambar 4.9 menjelaskan alur proses TF-IDF. Dimulai dari masukan berupa hasil dari proses *stemming*. Setiap kata pada hasil *stemming* akan dihitung dan menjadi nilai TF. Selanjutnya menghitung nilai DF dan IDF yang kemudian akan disimpan pada kolom disebelah nilai TF. Proses selanjutnya yaitu menghitung TF-IDF yang didapat dari hasil perkalian antara nilai tf masing-masing term dengan nilai IDF. Hasil dari proses TF-IDF akan digunakan pada proses selanjutnya.

4.4.7 Cosine Similarity

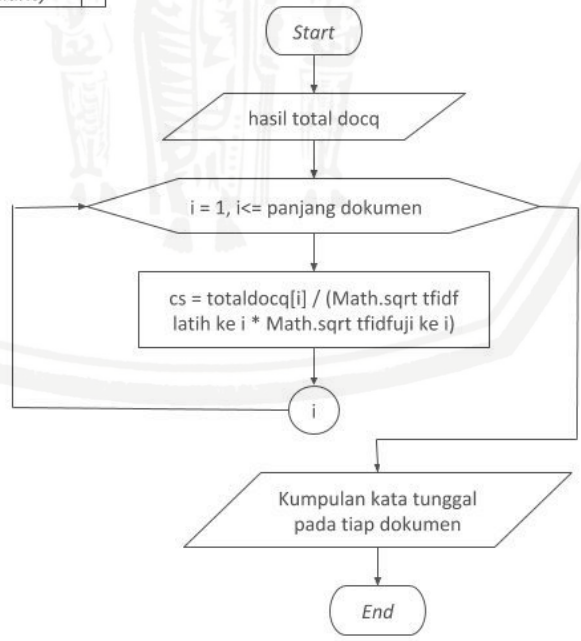
Tahap terakhir dari Text Similarity yaitu *cosine similarity*. *Cosine similarity* merupakan salah satu metode untuk menghitung nilai kesamaan antara dua dokumen dengan mencari nilai kemiripan dari dua buah dokumen. Hasil dari *cosine similarity* menunjukkan tingkat kemiripan antara dokumen satu dengan dokumen yang lainnya. Penjelasan alir proses *Cosine Similarity* dijabarkan dalam Gambar 4.10



Cosine Similarity



Cosine Similarity



Gambar 4. 10 Diagram alir proses Cosine Similarity



Gambar 4.10 menjelaskan alur proses *Cosine Similarity*. Dimulai dari masukan berupa hasil dari proses TFIDF. Kemudian dilakukan perkalian antara hasil TFIDF data latih dengan TFIDF dari data uji. Selanjutnya hasil dari perkalian tersebut akan ditranspose. Setelah dilakukan transpose maka nilainya akan dijumlahkan. Selanjutnya dilakukan perhitungan *cosine similarity* dengan melakukan pembagian antara hasil TFIDF yang telah ditranspose dengan akar dari TFIDF latih yang dikalikan dengan akar dari TFIDF uji. Hasil dari perhitungan *cosine similarity* ini merupakan nilai yang menunjukkan kedekatan dokumen. Semakin tinggi nilai *cosine similarity* maka akan semakin mirip dengan dokumen yang dituju.

4.5 Perhitungan Manual

Tujuan dari perhitungan manual dalam penelitian yaitu untuk mempermudah menjelaskan mengenai gambaran umum mengenai perancangan sistem prediksi rating novel berdasarkan sinopsis berbahasa Inggris dengan menggunakan *Genre Based Collaborative Filtering* dan *Text Similarity*. Pada perhitungan manual digunakan tiga sinopsis novel yang didapat dari situs *goodreads.com* sebagai data latih dan satu sinopsis novel yang didapat dari situs *goodreads.com* sebagai data uji. Berikut ini merupakan perhitungan manual:

1. Genre Based Collaborative Filtering

Proses selanjutnya yaitu perhitungan prediksi rating dengan menggunakan *genre based collaborative filtering*. Langkah awal dalam melakukan prediksi rating yaitu dengan menuliskan data matriks dari *user-genre* rating. Nilai rating didapatkan dari rating yang diberikan oleh *user* pada novel. Data matriks ditunjukkan pada Tabel 4.1.

Tabel 4. 1 Data matriks *user-genre* rating

User	Genre		
	Fiction	Horror	Sci-fic
1	4	4	1
2	?	3	4
3	3	?	3

Kemudian melakukan perhitungan pencarian nilai rata-rata *user*. Nilai rata-rata *user* digunakan dalam menghitung kesamaan *genre-to-genre* dengan menggunakan *cosine similarity* seperti pada persamaan 2.1. Untuk genre yang tidak diberikan rating oleh *user*, maka dihitung 0. Nilai rata-rata *user* ditunjukkan pada tabel 4.2.

Tabel 4. 2 Nilai rata-rata *user*

User	Genre			Rata-rata <i>user</i>
	Fiction	Horror	Sci-fic	
1	4	4	1	$\bar{R}_u = \frac{4 + 4 + 1}{3} = 3$
2	?	3	4	$\bar{R}_u = \frac{0 + 3 + 4}{3} = 2,333$
3	3	?	3	$\bar{R}_u = \frac{3 + 0 + 3}{3} = 2$

Tabel 4.3 menunjukkan matriks kesamaan *genre-to-genre* dari hasil perhitungan dengan menggunakan *cosine similarity* seperti pada persamaan 2.1.

Tabel 4. 3 Matriks kesamaan *genre-to-genre*

Kesamaan <i>genre-to-genre</i>	pembilang	penyebut	$\frac{\text{pembilang}}{\text{penyebut}}$
Fiksi - Horror	$(4 - 3) \times (4 - 3) = 1$	$\sqrt{(4 - 3)^2} \times \sqrt{(4 - 3)^2} = 1$	$\frac{1}{1} = 1$
Fiksi - Sci-fic	$((4 - 3) \times (1 - 3)) + ((3 - 2) \times (3 - 2)) = -1$	$\sqrt{(4 - 3)^2 + (3 - 2)^2} \times \sqrt{(1 - 3)^2 + (3 - 2)^2} = 6$	$\frac{-1}{6} = -0,167$
Horror - Sci-fic	$((4 - 3) \times (1 - 3)) + ((3 - 2,333) \times (4 - 2,333)) = -0,889$	$\sqrt{(4 - 3)^2 + (3 - 2,333)^2} \times \sqrt{(1 - 3)^2 + (4 - 2,333)^2} = 6,901$	$\frac{-0,889}{6,901} = -0,128$

Setelah diketahui kesamaan *genre-to-genre* dengan *cosine similarity* selanjutnya yaitu melakukan perhitungan *rating* yang tidak dirating oleh *user* dengan menggunakan persamaan 2.2. Setelah diketahui nilai *rating* yang tidak dirating oleh *user* maka matriks *user-genre* menjadi pada Tabel 4.4.

Tabel 4. 4 Matriks *user-genre* rating

User	Genre		
	Fiction	Horror	Sci-fic
1	4	4	1
2	$\frac{(3 \times -0,167) + (4 \times -0,167)}{-0,167} = 7$	3	4
3	3	$\frac{(3 \times -0,128) + (3 \times -0,128)}{-0,128} = 6$	3

Nilai maksimal pada rating merupakan 5 sehingga nilai genre yang kosong menjadi pada Tabel 4.5.

Tabel 4. 5 Hasil prediksi rating

User	Genre		
	Fiction	Horror	Sci-fic
1	4	4	1
2	5	3	4
3	3	5	3

2. Dokumen sinopsis awal

Tabel 4.6 menunjukkan dokumen sinopsis awal yang merupakan dokumen masukan yang kemudian diolah menjadi dokumen yang lebih terstruktur yang berfungsi untuk mengetahui bobot masing-masing term dengan menggunakan TFIDF serta untuk menghitung nilai kesamaan dokumen menggunakan *cosine similarity*.

Tabel 4. 6 Dokumen sinopsis novel

Sinopsis	Genre	Rating
Everyone in Shaker Heights was talking about it that summer: how Isabelle, the last of the Richardson children, had finally gone around the bend and burned the house down. In Shaker Heights, a placid, progressive suburb of Cleveland, everything is meticulously planned – from the layout of the winding roads, to the colours of the houses, to the successful lives its residents will go on to lead. And no one embodies this spirit more than Elena Richardson, whose guiding principle is playing by the rules. Enter Mia Warren – an enigmatic artist and single mother – who arrives in this idyllic bubble with her teenage daughter Pearl, and rents a house from the Richardsons. Soon Mia and Pearl become more than just tenants: all four Richardson children are drawn to the alluring mother-daughter pair. But Mia carries with her a mysterious past, and a disregard for the rules that threatens to upend this carefully ordered community. When the Richardsons' friends attempt to adopt a Chinese-American baby, a custody battle erupts that dramatically divides the town and puts Mia and Mrs. Richardson on opposing sides. Suspicious of Mia and her motives, Mrs. Richardson becomes determined to uncover the secrets in Mia's past. But her obsession will come at unexpected and devastating costs to her own family – and Mia's. <i>Little Fires Everywhere</i> explores the weight of long-held secrets and the ferocious pull of motherhood-and the danger of believing that	Fiction	4,18

planning and following the rules can avert disaster, or heartbreak. (Ng, 2017)		
In a future so real and near it might be now, something happens when women go to sleep; they become shrouded in a cocoon-like gauze. If they are awakened, if the gauze wrapping their bodies is disturbed or violated, the women become feral and spectacularly violent; and while they sleep they go to another place. The men of our world are abandoned, left to their increasingly primal devices. One woman, however, the mysterious Evie, is immune to the blessing or curse of the sleeping disease. Is Evie a medical anomaly to be studied, or is she a demon who must be slain? (King & King, 2017)	Horror	3,8
Jazz Bashara is a criminal. Well, sort of. Life on Artemis, the first and only city on the moon, is tough if you're not a rich tourist or an eccentric billionaire. So smuggling in the occasional harmless bit of contraband barely counts, right? Not when you've got debts to pay and your job as a porter barely covers the rent. Everything changes when Jazz sees the chance to commit the perfect crime, with a reward too lucrative to turn down. But pulling off the impossible is just the start of her problems, as she learns that she's stepped square into a conspiracy for control of Artemis itself—and that now, her only chance at survival lies in a gambit even riskier than the first. (Weir, 2017)	Science-fiction	3,69
After getting a note demanding his presence, Federal Agent Aaron Falk arrives in his hometown for the first time in decades to attend the funeral of his best friend, Luke. Twenty years ago when Falk was accused of murder, Luke was his alibi. Falk and his father fled under a cloud of suspicion, saved from prosecution only because of Luke's steadfast claim that the boys had been together at the time of the crime. But now more than one person knows they didn't tell the truth back then, and Luke is dead. Amid the worst drought in a century, Falk and the local detective question what really happened to Luke. As Falk reluctantly investigates to see if there's more to Luke's death than there seems to be, long-buried mysteries resurface, as do the lies that have haunted them. And Falk will find that small towns have always hidden big secrets. (Harper, 2016)	?	?

3. Tokenization

Tabel 4.7 menunjukkan dokumen yang sudah dilakukan *tokenization* atau pemotongan *string input* atau memecah kalimat menjadi kata.

Tabel 4. 7 Manualisasi *Tokenization*

Sinopsis	Genre	Rating
Everyone // in // Shaker // Heights // was // talking // about // it // that // summer:// how // Isabelle,// the // last // of // the // Richardson // children, // had // finally // gone // around // the //	Fiction	4,18

<p>bend // and // burned // the // house // down. // In // Shaker // Heights, // a // placid, // progressive // suburb // of // Cleveland, // everything // is // meticulously // planned // – // from // the // layout // of // the // winding // roads, // to // the // colours // of // the // houses, // to // the // successful // lives // its // residents // will // go // on // to // lead. // And // no // one // embodies // this // spirit // more // than // Elena // Richardson, // whose // guiding // principle // is // playing // by // the // rules. // Enter // Mia // Warren // – // an // enigmatic // artist // and // single // mother // – // who // arrives // in // this // idyllic // bubble // with // her // teenage // daughter // Pearl, // and // rents // a // house // from // the // Richardsons. // Soon // Mia // and // Pearl // become // more // than // just // tenants: // all // four // Richardson // children // are // drawn // to // the // alluring // mother // - // daughter // pair. // But // Mia // carries // with // her // a // mysterious // past, // and // a // disregard // for // the // rules // that // threatens // to // upend // this // carefully // ordered // community. // When // the // Richardsons' // friends // attempt // to // adopt // a // Chinese // - // American // baby, // a // custody // battle // erupts // that // dramatically // divides // the // town // and // puts // Mia // and // Mrs. // Richardson // on // opposing // sides. // Suspicious // of // Mia // and // her // motives, // Mrs. // Richardson // becomes // determined // to uncover // the // secrets // in // Mia's // past. // But // her // obsession // will // come // at // unexpected // and // devastating // costs // to // her // own // family // – // and // Mia's. // Little // Fires // Everywhere // explores // the // weight // of // long // -// held // secrets // and // the // ferocious // pull // of // motherhood // - // and // the // danger // of // believing // that // planning // and // following // the // rules // can // avert // disaster, // or // heartbreak. //</p>		
<p>In // a // future // so // real // and // near // it // might // be // now, // something // happens // when // women // go // to // sleep; // they // become // shrouded // in // a // cocoon // - // like // gauze. // If // they // are // awakened, // if // the // gauze // wrapping // their // bodies // is // disturbed // or // violated, // the // women // become // feral // and // spectacularly // violent; // and // while // they // sleep // they // go // to // another // place. // The // men // of // our // world // are // abandoned, // left // to // their // increasingly // primal // devices. // One // woman, // however, // the // mysterious // Evie, // is // immune // to // the // blessing // or // curse // of // the // sleeping // disease. // Is // Evie // a // medical // anomaly // to // be // studied, // or // is // she // a // demon // who // must // be // slain? //</p>	Horror	3,8
<p>Jazz // Bashara // is // a // criminal. // Well, // sort // of. // Life // on // Artemis, // the // first // and // only // city // on // the // moon, // is // tough // if // you're // not // a // rich // tourist // or // an // eccentric // billionaire. // So // smuggling // in // the //</p>	Science-fiction	3,69



<p>occasional // harmless // bit // of // contraband // barely // counts, // right? // Not // when // you've // got // debts // to // pay // and // your // job // as // a // porter // barely // covers // the // rent. // Everything // changes // when // Jazz // sees // the // chance // to // commit // the // perfect // crime, // with // a // reward // too // lucrative // to // turn // down. // But // pulling // off // the // impossible // is // just // the // start // of // her // problems, // as // she // learns // that // she's // stepped // square // into // a // conspiracy // for // control // of // Artemis // itself // — // and // that // now, // her // only // chance // at // survival // lies // in // a // gambit // even // riskier // than // the // first. //</p>		
<p>After // getting // a // note // demanding // his // presence, // Federal // Agent // Aaron // Falk // arrives // in // his // hometown // for // the // first // time // in // decades // to // attend // the // funeral // of // his // best // friend, // Luke. // Twenty // years // ago // when // Falk // was // accused // of // murder, // Luke // was // his // alibi. // Falk // and // his // father // fled // under // a // cloud // of // suspicion, // saved // from // prosecution // only // because // of // Luke's // steadfast // claim // that // the // boys // had // been // together // at // the // time // of // the // crime. // But // now // more // than // one // person // knows // they // didn't // tell // the // truth // back // then, // and // Luke // is // dead. // Amid // the // worst // drought // in // a // century, // Falk // and // the // local // detective // question // what // really // happened // to // Luke. // As // Falk // reluctantly // investigates // to // see // if // there's // more // to // Luke's // death // than // there // seems // to // be, // long // - // buried // mysteries // resurface, // as // do // the // lies // that // have // haunted // them. // And // Falk // will // find // that // small // towns // have // always // hidden // big // secrets.//</p>	?	?

4. *Cleaning*

Tabel 4.8 menunjukkan dokumen yang sudah dilakukan *cleaning* atau penghapusan tanda baca.

Tabel 4. 8 Manualisasi *Cleaning*

Sinopsis	Genre	Rating
<p>Everyone // in // Shaker // Heights // was // talking // about // it // that // summer // how // Isabelle// the // last // of // the // Richardson // children // had // finally // gone // around // the // bend // and // burned // the // house // down // In // Shaker // Heights // a // placid // progressive // suburb // of // Cleveland // everything // is // meticulously // planned // from // the // layout // of // the // winding // roads // to // the // colours // of // the // houses // to // the // successful // lives // its // residents // will // go // on // to // lead // And // no // one // embodies // this // spirit // more // than // Elena // Richardson // whose // guiding</p>	Fiction	4,18



<p>// principle // is // playing // by // the // rules // Enter // Mia // Warren // an // enigmatic // artist // and // single // mother // who // arrives // in // this // idyllic // bubble // with // her // teenage // daughter // Pearl // and // rents // a // house // from // the // Richardsons // Soon // Mia // and // Pearl // become // more // than // just // tenants // all // four // Richardson // children // are // drawn // to // the // alluring // mother // daughter // pair // But // Mia // carries // with // her // a // mysterious // past // and // a // disregard // for // the // rules // that // threatens // to // upend // this // carefully // ordered // community // When // the // Richardsons // friends // attempt // to // adopt // a // Chinese// American // baby // a // custody // battle // erupts // that // dramatically // divides // the // town // and // puts // Mia // and // Mrs // Richardson // on // opposing // sides // Suspicious // of //Mia // and // her // motives // Mrs // Richardson // becomes // determined // to uncover // the // secrets // in // Mias // past // But // her // obsession // will // come // at // unexpected // and // devastating // costs // to // her // own // family // and // Mias // Little // Fires // Everywhere // explores // the // weight // of // long // held // secrets // and // the // ferocious // pull // of // motherhood // and // the // danger // of // believing // that // planning // and // following // the // rules // can // avert // disaster // or // heartbreak //</p>		
<p>In // a // future // so // real // and // near // it // might // be // now // something // happens // when // women // go // to // sleep // they // become // shrouded // in // a // cocoon // like // gauze // If // they // are // awakened // if // the // gauze // wrapping // their // bodies // is // disturbed // or // violated // the // women // become // feral // and // spectacularly // violent // and // while // they // sleep // they // go // to // another // place // The // men // of // our // world // are // abandoned // left // to // their // increasingly // primal // devices. // One // woman // however // the // mysterious // Evie // is //immune // to // the // blessing // or // curse // of // the // sleeping // disease // Is // Evie // a // medical // anomaly // to // be // studied // or // is // she // a // demon // who // must // be // slain //</p>	Horror	3,8
<p>Jazz // Bashara // is // a // criminal // Well // sort // of // Life // on // Artemis // the // first // and // only // city // on // the // moon // is // tough // if // youre // not // a // rich // tourist // or // an // eccentric // billionaire // So // smuggling // in // the // occasional // harmless // bit // of // contraband // barely // counts // right // Not // when // youve // got // debts // to // pay // and // your // job // as // a // porter // barely // covers // the // rent // Everything // changes // when // Jazz // sees // the // chance // to // commit // the // perfect // crime // with // a // reward // too // lucrative // to // turn // down // But // pulling // off // the // impossible // is // just // the // start //of // her // problems // as // she // learns // that // shes // stepped // square // into // a // conspiracy // for // control // of // Artemis // itself //</p>	Science-fiction	3,69



// and // that // now // her // only // chance // at // survival // lies // in // a // gambit // even // riskier // than // the // first //		
After // getting // a // note // demanding // his // presence // Federal // Agent // Aaron // Falk // arrives // in // his // hometown // for // the // first // time // in // decades // to // attend // the // funeral // of // his // best // friend // Luke // Twenty // years // ago // when // Falk // was // accused // of // murder // Luke // was // his // alibi // Falk // and // his // father // fled // under // a // cloud // of // suspicion // saved // from // prosecution // only // because // of // Lukes // steadfast // claim // that // the // boys // had // been // together // at // the // time // of // the // crime // But // now // more // than // one // person // knows // they // didnt // tell // the // truth // back // then, // and // Luke // is // dead. // Amid // the // worst // drought // in // a // century // Falk // and // the // local // detective // question // what // really // happened // to // Luke // As // Falk // reluctantly // investigates // to // see // if // theres // more // to // Lukes // death // than // there // seems // to // be // long // buried // mysteries // resurface // as // do // the // lies // that // have // haunted // them // And // Falk // will // find // that // small // towns // have // always // hidden // big // secrets//	?	?

5. Case Folding

Tabel 4.9 menunjukkan hasil dokumen yang telah dilakukan perubahan semua huruf pada dokumen menjadi *lower case*.

Tabel 4. 9 Manualisasi Case Folding

Sinopsis	Genre	Rating
everyone // in // shaker // heights // was // talking // about // it // that // summer // how // isabelle// the // last // of // the // richardson // children // had // finally // gone // around // the // bend // and // burned // the // house // down // in // shaker // heights // a // placid // progressive // suburb // of // cleveland // everything // is // meticulously // planned // from // the // layout // of // the // winding // roads // to // the // colours // of // the // houses // to // the // successful // lives // its // residents // will // go // on // to // lead // and // no // one // embodies // this // spirit // more // than // elena // richardson // whose // guiding // principle // is // playing // by // the // rules // enter // mia // warren // an // enigmatic // artist // and // single // mother // who // arrives // in // this // idyllic // bubble // with // her // teenage // daughter // pearl // and // rents // a // house // from // the // richardsons // soon // mia // and // pearl // become // more // than // just // tenants // all // four // richardson // children // are // drawn // to // the // alluring // mother // daughter // pair // but // mia // carries // with // her // a // mysterious // past // and // a // disregard // for // the // rules // that // threatens // to // upend // this // carefully // ordered //	Fiction	4,18



<p>community // when // the // richardsons // friends // attempt // to // adopt // a // chinese// american // baby // a // custody // battle // erupts // that // dramatically // divides // the // town // and // puts // mia // and // mrs // richardson // on // opposing // sides // suspicious // of //mia // and // her // motives // mrs // richardson // becomes // determined // to uncover // the // secrets // in // mias // past // but // her // obsession // will // come // at // unexpected // and // devastating // costs // to // her // own // family // and // mias // little // fires // everywhere // explores // the // weight // of // long // held // secrets // and // the // ferocious // pull // of // motherhood // and // the // danger // of // believing // that // planning // and // following // the // rules // can // avert // disaster // or // heartbreak //</p>		
<p>in // a // future // so // real // and // near // it // might // be // now // something // happens // when // women // go // to // sleep // they // become // shrouded // in // a // cocoon // like // gauze // if // they // are // awakened // if // the // gauze // wrapping // their // bodies // is // disturbed // or // violated // the // women // become // feral // and // spectacularly // violent // and // while // they // sleep // they // go // to // another // place // the // men // of // our // world // are // abandoned // left // to // their // increasingly // primal // devices. // one // woman // however // the // mysterious // evie // is //immune // to // the // blessing // or // curse // of // the // sleeping // disease // is // evie // a // medical // anomaly // to // be // studied // or // is // she // a // demon // who // must // be // slain //</p>	Horror	3,8
<p>jazz // bashara // is // a // criminal // well // sort // of // life // on // artemis // the // first // and // only // city // on // the // moon // is // tough // if // youre // not // a // rich // tourist // or // an // eccentric // billionaire // so // smuggling // in // the // occasional // harmless // bit // of // contraband // barely // counts // right // not // when // youve // got // debts // to // pay // and // your // job // as // a // porter // barely // covers // the // rent // everything // changes // when // jazz // sees // the // chance // to // commit // the // perfect // crime // with // a // reward // too // lucrative // to // turn // down // but // pulling // off // the // impossible // is // just // the // start //of // her // problems // as // she // learns // that // shes // stepped // square // into // a // conspiracy // for // control // of // artemis // itself // and // that // now // her // only // chance // at // survival // lies // in // a // gambit // even // riskier // than // the // first //</p>	Science-fiction	3,69
<p>after // getting // a // note // demanding // his // presence // federal // agent // aaron // falk // arrives // in // his // hometown // for // the // first // time // in // decades // to // attend // the // funeral // of // his // best // friend // luke // twenty // years // ago // when // falk // was // accused // of // murder // luke // was // his // alibi // falk // and // his // father // fled // under // a // cloud // of // suspicion // saved // from // prosecution // only // because // of // lukes // steadfast // claim // that // the // boys //</p>	?	?



had // been // together // at // the // time // of // the // crime // but // now // more // than // one // person // knows // they // didnt // tell // the // truth // back // then, // and // luke // is // dead. // amid // the // worst // drought // in // a // century // falk // and // the // local // detective // question // what // really // happened // to // luke // as // falk // reluctantly // investigates // to // see // if // theres // more // to // lukes // death // than // there // seems // to // be // long // buried // mysteries // resurface // as // do // the // lies // that // have // haunted // them // and // falk // will // find // that // small // towns // have // always // hidden // big // secrets//		
--	--	--

6. Filtering

Dalam proses *filtering* dilakukan penghapusan kata yang sama seperti pada kamus *stopword*. Tabel 4.10 menunjukkan dokumen yang telah dilakukan *filtering*.

Tabel 4. 10 Manualisasi *Filtering*

Sinopsis	Genre	Rating
shaker // heights // talking // summer // isabelle // richardson // children // finally // bend // burned // house // shaker // heights // placid // progressive // suburb // cleveland // meticulously // planned // layout // winding // roads // colours // houses // successful // lives // residents // lead // embodies // spirit // elena // richardson // guiding // principle // playing // rules // enter // mia // warren // enigmatic // artist // single // mother // arrives // idyllic // bubble // teenage // daughter // pearl // rents // house // richardsons // mia // pearl // tenants // richardson // children // drawn // alluring // mother // daughter // pair // mia // carries // mysterious // past // disregard // rules // threatens // upend // carefully // ordered // community // richardsons // friends // attempt // adopt // chinese// american // baby // custody // battle // erupts // dramatically // divides // town // richardson // opposing // sides // suspicious // mia // motives // mrs // richardson // becomes // determined // uncover // secrets // mias // past // obsession // unexpected // devastating // costs // own // family // mias // fires // explores // weight // long // held // secrets // ferocious // pull // motherhood // danger // believing // planning // following // rules // avert // disaster // heartbreak //	Fiction	4,18
future // real // women // sleep // shrouded // cocoon // gauze // awakened // gauze // wrapping // bodies // disturbed // violated // women // feral // spectacularly // violent // sleep // place // men // world // abandoned // left // increasingly // primal // devices // woman // mysterious // evie // immune // blessing // curse // sleeping // disease // evie // medical // anomaly // studied // demon // slain //	Horror	3,8



jazz // bashara // criminal // sort // life // artemis // city // moon // tough // rich // tourist // eccentric // billionaire // smuggling // occasional // harmless // bit // contraband // barely // counts // debts // pay // job // porter // barely // covers // rent // changes // jazz // sees // chance // commit // perfect // crime // reward // lucrative // turn // pulling // impossible // start // problems // learns // stepped // square // conspiracy // control // artemis // chance // survival // lies // gambit // riskier //	Science-fiction	3,69
note // demanding // presence // federal // agent // aaron // falk // arrives // hometown // time // decades // attend // funeral // friend // luke // twenty // years // ago // falk // accused // murder // luke // alibi // falk // father // fled // cloud // suspicion // saved // prosecution // lukes // steadfast // claim // boys // time // crime // person // knows // tell // truth // luke // dead // amid // worst // drought // century // falk // local // detective // question // happened // luke // falk // reluctantly // investigates // lukes // death // long // buried // mysteries // resurface // lies // haunted // falk // find // small // towns // hidden // big // secrets//	?	?

7. *Stemming*

Tabel 4.11 menunjukkan dokumen yang telah di-*stemming* atau telah dilakukan penghapusan kata yang berimbuhan sehingga hanya tersisa kata dasar.

Tabel 4. 11 Manualisasi *Stemming*

Sinopsis	Genre	Rating
shaker // height // talk // summer // isabelle // richardson // children // finally // bend // burn // house // shaker // height // placid // progress // suburb // cleveland // meticulous // plan // layout // wind // road // colour // house // success // live // resident // lead // embody // spirit // elena // richardson // guide // principle // play // rule // enter // mia // warren // enigmatic // artist // single // mother // arrive // idyllic // bubble // teenage // daughter // pearl // rent // house // richardson // mia // pearl // tenant // richardson // children // drawn // allure // mother // daughter // pair // mia // carry // mystery // past // disregard // rule // threaten // upend // careful // order // community // richardson // friend // attempt // adopt // chinese // american // baby // custody // battle // erupt // dramatical // divide // town // richardson // oppose // side // suspici // mia // motive // mrs // richardson // become // determine // uncover // secret // mia // past // obsession // unexpect // devastat // cost // own // family // mia // fire // explore // weight // long // held // secret // feroci // pull // motherhood // danger // believe // plan // follow // rule // avert // disaster // heartbreak //	Fiction	4,18
future // real // women // sleep // shroud // cocoon // gauze // awaked // gauze // wrap // body // disturb // violat // women // feral // spectacular // violent // sleep // place // men // world // abandon // left // increasing // primal // device // woman //	Horror	3,8



mystery // evie // immune // bless // curse // sleep // disease // evie // medical // anomaly // study // demon // slain //		
jazz // bashara // criminal // sort // life // artemis // city // moon // tough // rich // tourist // eccentric // billionaire // smuggle // occasional // harmless // bit // contraband // bare // count // debt // pay // job // porter // bare // cover // rent // change // jazz // see // chance // commit // perfect // crime // reward // lucrative // turn // pull // impossible // start // problem // learn // step // square // conspiracy // control // artemis // chance // survival // lie // gambit // risk //	Science-fiction	3,69
note // demand // presence // federal // agent // aaron // falk // arrive // hometown // time // decade // attend // funeral // friend // luke // twenty // year // ago // falk // accuse // murder // luke // alibi // falk // father // fled // cloud // suspicion // save // prosecution // luke // steadfast // claim // boy // time // crime // person // know // tell // truth // luke // dead // amid // worst // drought // century // falk // local // detective // question // happen // luke // falk // reluctant // investigate // luke // death // long // bury // mystery // resurface // lie // haunt // falk // find // small // town // hidden // big // secret //	?	?

8. TF-IDF

Proses selanjutnya yaitu proses perhitungan bobot dengan menggunakan TF-IDF. Tabel 4.12 menunjukkan proses awal pemberian bobot pada term yang masih berupa *binary* TF dan *raw* TF. *Binary* TF menunjukkan terdapat atau tidaknya term pada dokumen, sedangkan *raw* TF menunjukkan jumlah yang sesungguhnya term pada dokumen.

Tabel 4. 12 Manualisasi *Binary* TF dan *Raw* TF

Term	Binary TF				Raw TF			
	Fiction	Horror	Sci-fic	?	Fiction	Horror	Sci-fic	?
shaker	1	0	0	0	2	0	0	0
height	1	0	0	0	2	0	0	0
talk	1	0	0	0	1	0	0	0
summer	1	0	0	0	1	0	0	0
isabelle	1	0	0	0	1	0	0	0
richardson	1	0	0	0	7	0	0	0
children	1	0	0	0	2	0	0	0
final	1	0	0	0	1	0	0	0
...
lie	0	0	1	1	0	0	1	1



gambit	0	0	1	0	0	0	1	0
riskier	0	0	1	0	0	0	1	0

Setelah mengetahui *binary* TF dan *raw* TF, maka dilakukan perhitungan log TF terhadap raw TF. Setelah itu dilakukan perhitungan untuk mengetahui IDF dengan cara melakukan pembagian pada jumlah dokumen sebagai data latih dengan jumlah dari binary TF pada data latih. Hasil dari pembagian tersebut kemudian di-*log* sehingga menghasilkan IDF. Hasil dari perhitungan log TF dan IDF ditunjukkan pada Tabel 4.13.

Tabel 4. 13 Manualisasi Log TF dan IDF

Term	Log TF				IDF		
	Fiction	Horror	Sci-fic	?	DF	N/DF	IDF
shaker	1,30103	0	0	0	1	3	0,477121
height	1,30103	0	0	0	1	3	0,477121
talk	1	0	0	0	1	3	0,477121
summer	1	0	0	0	1	3	0,477121
isabelle	1	0	0	0	1	3	0,477121
richardson	1,845098	0	0	0	1	3	0,477121
children	1,30103	0	0	0	1	3	0,477121
final	1	0	0	0	1	3	0,477121
...
lie	0	0	1	1	1	3	0,477121
gambit	0	0	1	0	1	3	0,477121
riskier	0	0	1	0	1	3	0,477121

Setelah diketahui *log* TF dan IDF, selanjutnya adalah menghitung nilai TF-IDF yang kemudian menjadi bobot dari *term* tersebut. Nilai TF-IDF didapatkan dari hasil perkalian antara *log* TF dengan IDF. Hasil dari perhitungan TF-IDF ditunjukkan dalam Tabel 4.14.

Tabel 4. 14 Manualisasi TF-IDF

term	TF-IDF			
	Fiction	Horror	Sci-fic	?
shaker	0,620749	0	0	0
height	0,620749	0	0	0

term	TF-IDF			
	Fiction	Horror	Sci-fic	?
talk	0,477121	0	0	0
summer	0,477121	0	0	0
isabelle	0,477121	0	0	0
richardson	0,880335	0	0	0
children	0,620749	0	0	0
final	0,477121	0	0	0
...
lie	0	0	0,477121	0,477121
gambit	0	0	0,477121	0
riskier	0	0	0,477121	0

9. Cosine Similarity

Dalam menghitung *cosine similarity*, diperlukan melakukan kuadrat pada hasil TF-IDF dan perkalian antara masing-masing TF-IDF data latih dengan TF-IDF data uji. Kemudian masing-masing hasil kuadrat TF-IDF pada data latih dan data uji dijumlahkan. Pada perkalian TF-IDF data latih dengan TF-IDF data uji dijumlahkan yang dijadikan sebagai vektor dari *A* dan *B*. Setelah itu dilakukan perkalian antara jumlah dari kuadrat TF-IDF pada data uji dengan masing-masing jumlah dari kuadrat TF-IDF pada data latih yang selanjutnya dijadikan sebagai panjang vektor *A* dan panjang vektor *B*. Setelah itu hasil dari *cosine similarity* didapatkan dari hasil pembagian antara vektor *A* dan *B* dengan panjang vektor *A* dan panjang vektor *B*. Perhitungan tersebut menggunakan rumus pada persamaan 2.4. Hasil manualisasi dari *cosine similarity* ditunjukkan pada Tabel 4.15.

Tabel 4. 15 Manualisasi Cosine Similarity

	Fiction	Horror	Sci-fic
vektor x.y	1,237759	0,031008	0,455289
panjang vektor x dan y	5,015631	2,91771	3,307078
<i>Cosine similarity</i>	0,193619	0,008338	0,108014

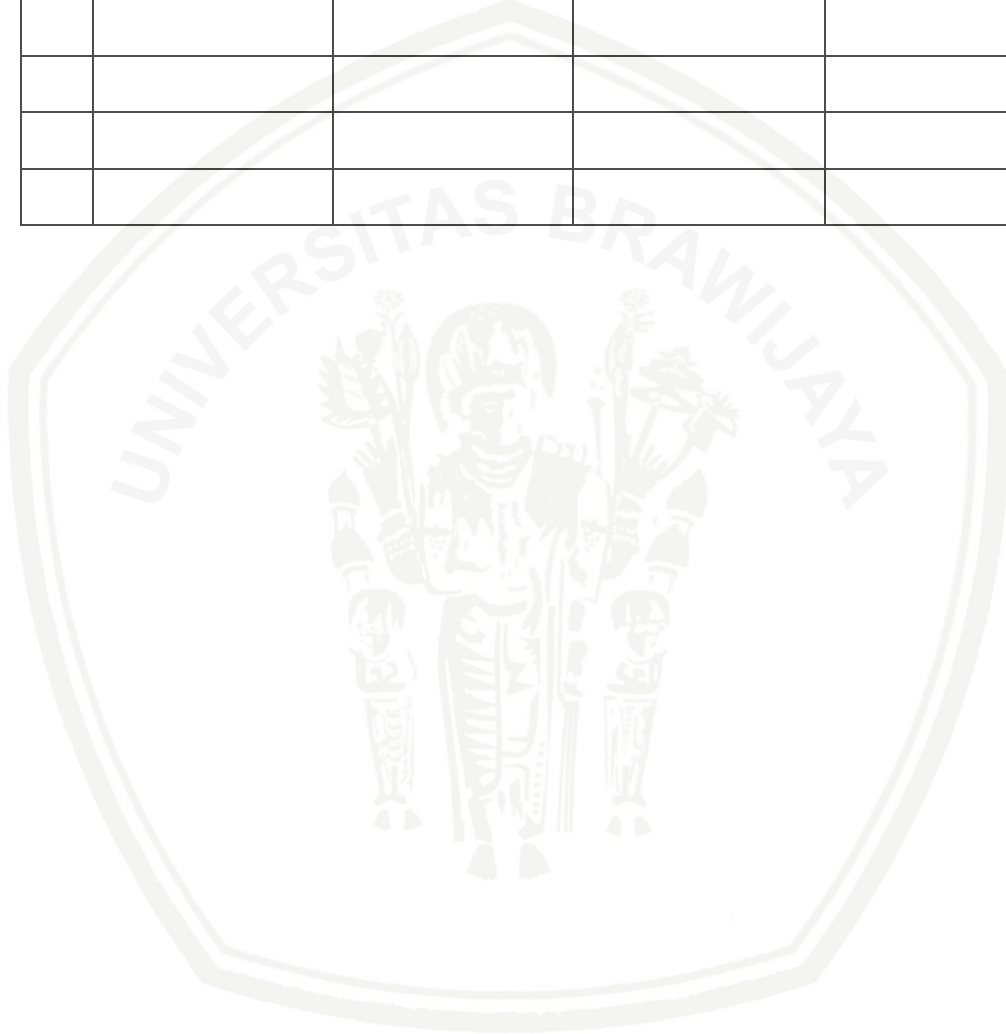
Dari hasil *cosine similarity*, dokumen data latih “Fiction” memiliki nilai *cosine similarity* tertinggi dengan nilai 0,193619, dan dapat disimpulkan bahwa kecenderungan kesamaan genre pada dokumen data uji “?” mendekati dokumen data latih pertama yaitu *fiction*.

4.6 Perancangan Pengujian

Perancangan pengujian digunakan untuk merancang pengujian dalam melihat seberapa baik hasil akurasi dari metode *genre based collaborative filtering* dan *text similarity* yang dihasilkan dengan memasukkan data latih dan data uji ke dalam sistem.

Tabel 4. 16 Tabel perancangan pengujian

No	Genre Asli	Rating Asli	Genre Prediksi	Rating Prediksi



BAB 5 IMPLEMENTASI

5.1 Implementasi Sistem

Pada implementasi sistem akan dijelaskan tahapan dalam melakukan prediksi rating novel baru berdasarkan sinopsis dengan menggunakan metode *genre based collaborative filtering* dan *text similarity*. Tahapan tersebut antara lain *tokenization, cleaning, case folding, filtering, stemming*, penghitungan TF-IDF, penghitungan *cosine similarity*, dan penghitungan rating dengan *genre based collaborative filtering*.

5.1.1 Genre Based Collaborative Filtering

Salah satu metode yang memberikan hasil rekomendasi berdasarkan kemiripan antar genre yaitu metode *genre based collaborative filtering*. Dalam proses *genre based collaborative filtering* dilakukan analisis mengenai hubungan antar genre dengan user dengan matriks *user-genre*. Proses prediksi menggunakan *Genre Based Collaborative Filtering* ditunjukkan pada Kode Program 5.1.

```

1 public double similarityGenre(int[][] a, int nbaris, int j,
2   int k, float[] Ru) {
3     for (int i = 1; i < nbaris; i++) {
4       if (a[i][j] != 0 && a[i][k] != 0) {
5         pb += ((a[i][j] - Ru[i]) * (a[i][k] - Ru[i]));
6         py1 += ((a[i][j] - Ru[i]) * (a[i][j] -
7   Ru[i]));
8         py2 += ((a[i][k] - Ru[i]) * (a[i][k] -
9   Ru[i]));
10        }
11      }
12      py1sqrt = Math.sqrt(py1);
13      py2sqrt = Math.sqrt(py2);
14      acs = pb / (py1sqrt * py2sqrt);
15      kolommatriks[j][k] = (acs);
16      pb = 0;
17      py1 = 0;
18      py2 = 0;
19    }
20    return (acs);
21  }
22
23 public Double prediksiRating(int i, int j, List<Integer>
24   nilai_sim, int[][] a) {
25   pembilang = 0;
26   penyebut = 0;
27   for (int k = 0; k < nilai_sim.size(); k++) {
28     if (a[i][nilai_sim.get(k)] != 0) {
29       pembilang +=
30       ((kolommatriks[j][nilai_sim.get(k)])
31       *
32       (a[i][nilai_sim.get(k)]));

```

```

30         penyebut +=
Math.abs(kolommatriks[j][nilai_sim.get(k)]);
31     }
32 }
33     Double nilaiRating = pembilang / penyebut;
34     return nilaiRating;
35 }

```

Kode Program 5. 1 Prediksi rating dengan *genre based collaborative filtering*

Penjelasan Kode Program 5.1 adalah sebagai berikut:

1. Baris ke 2 hingga 8 menghitung nilai yang akan digunakan dalam *adjusted cosine similarity*.
2. Baris ke 13 menghitung nilai *adjusted cosine similarity*.
3. Baris ke 27 hingga 32 menghitung nilai yang akan digunakan dalam perhitungan prediksi rating.
4. Baris ke 33 menghitung nilai prediksi rating.

5.1.2 Tokenization

Tokenization merupakan proses memecah kalimat menjadi beberapa bagian. Implementasi pemecahan kalimat menjadi beberapa bagian ditunjukkan pada Kode Program 5.2.

```

1 public ArrayList<ArrayList<String>> tokenization(String[]
data) {
2     ArrayList<ArrayList<String>> hasilToken = new
ArrayList<ArrayList<String>>();
3
4     for (int i = 0; i < data.length; i++) {
5         String[] token = null;
6         token = data[i].split(" ");
7         ArrayList<String> baris = new
ArrayList<String>();
8         for (int j = 0; j < token.length; j++) {
9             baris.add(token[j]);
10        }
11        hasilToken.add(baris);
12    }
13    return hasilToken;
14 }

```

Kode Program 5. 2 Implementasi *tokenization*

Penjelasan Kode Program 5.2 adalah sebagai berikut:

1. Baris ke 6 melakukan pemecahan kalimat menjadi term dengan batas spasi.

5.1.3 Cleaning

Cleaning merupakan proses menghilangkan simbol-simbol serta menghapus kata yang hanya mengandung satu atau dua huruf. Implementasi proses *cleaning* ditunjukkan pada Kode Program 5.3.

```

1 public ArrayList<ArrayList<String>> clean(
  ArrayList<ArrayList<String>> tokenization) {
2     ArrayList<ArrayList<String>> cleaning = new
  ArrayList<>();
3
4     for (int i = 0; i < tokenization.size(); i++) {
5         for (int j = 0; j < tokenization.get(i).size();
6             j++) {
7             tokenization.get(i).set(j,
8             tokenization.get(i).get(j).replaceAll("[ \\-.,;:)(?!.'`_",
9             ""));
10            }
11        }
12        for (int i = 0; i < tokenization.size(); i++) {
13            for (int j = 0; j < tokenization.get(i).size();
14                j++) {
15                if (Pattern.matches("[A-Za-z]{1,2}",
16                    tokenization.get(i).get(j))) {
17                    tokenization.get(i).remove(j);
18                }
19            }
20        }
21        for (int i = 0; i < tokenization.size(); i++) {
22            if (tokenization.get(i).isEmpty()) {
23                tokenization.remove(i);
24            }
25        }
26        cleaning = copy(cleaning, tokenization);
27        return cleaning;
28    }

```

Kode Program 5. 3 Implementasi *cleaning*

Penjelasan Kode Program 5.3 adalah sebagai berikut:

1. Baris ke 4 hingga 9 dilakukan pergantian simbol-simbol dengan *whitespace*.
2. Baris ke 10 hingga 16 dilakukan penghapusan kata yang hanya mengandung satu atau dua huruf.
3. Baris ke 18 hingga 22 dilakukan penghapusan isi dari *array* yang tidak mengandung huruf.

5.1.4 Case Folding

Case folding merupakan suatu proses yang bertujuan untuk mengubah semua huruf pada dokumen menjadi huruf kecil atau yang sering disebut sebagai *lowercase*. Kode Program 5.4 menunjukkan implementasi dari proses *case folding*.

```

1 public ArrayList<ArrayList<String>>
  case_folding(ArrayList<ArrayList<String>> cleaning) {
2     ArrayList<ArrayList<String>> case_folding = new
  ArrayList<>();
3     for (int i = 0; i < cleaning.size(); i++) {
4         for (int j = 0; j < cleaning.get(i).size(); j++)
5             {

```

```

5         cleaning.get(i).set(j,
cleaning.get(i).get(j).toLowerCase());
6     }
7 }
8     case_folding = copy(case_folding, cleaning);
9     return case_folding;
10 }

```

Kode Program 5. 4 Implementasi *case folding*

Penjelasan Kode Program 5.4 adalah sebagai berikut:

1. Baris ke 4 hingga 8 dilakukan proses pengubahan huruf balok menjadi huruf kecil dari hasil *cleaning*.

5.1.5 Filtering

Pada tahap *filtering* dilakukan penghapusan kata-kata yang sama pada kata-kata yang terdapat pada daftar *stopword*. Kata-kata *stopword* didapat dari halaman website ranks.nl. Implementasi proses *filtering* ditunjukkan pada Kode Program 5.5.

```

1 public ArrayList<ArrayList<String>>
filtering(ArrayList<ArrayList<String>> case_folding) throws
FileNotFoundException, IOException {
2     ArrayList<ArrayList<String>> filtering = new
ArrayList<>();
3     ArrayList<String> stopwords = new ArrayList<>();
4     BufferedReader in = new BufferedReader(new
FileReader("C:\\Users\\Rhevitta\\Documents\\NetBeansProject
s\\Novel\\src\\Skripsi\\stopword.txt"));
5     String line = "";
6     while ((line = in.readLine()) != null) {
7         stopwords.add(line);
8     }
9     in.close();
10    for (int i = 0; i < case_folding.size(); i++) {
11        for (int j = 0; j < case_folding.get(i).size();
j++) {
12            for (int k = 0; k < stopwords.size(); k++) {
13                if
(case_folding.get(i).get(j).equals(stopwords.get(k)) && j >
0) {
14                    case_folding.get(i).remove(j);
15                    j--;
16                }
17            }
18        }
19    }
20
21    stopwords.clear();
22    filtering = copy(filtering, case_folding);
23    return filtering;
24 }

```

Kode Program 5. 5 Implementasi *Filtering*

Penjelasan Kode Program 5.5 adalah sebagai berikut:

1. Baris ke 4 dilakukan proses pembacaan file *stopword.txt*.

2. Baris ke 7 semua kata dalam stopwords.txt akan disimpan dalam *arraylist stopwords*.
3. Baris ke 10 hingga 19 dilakukan proses penghapusan kata *stopword*.

5.1.6 Stemming

Stemming merupakan proses pengubahan kata menjadi kata dasar dengan syarat-syarat tertentu. Jenis *stemming* yang digunakan yaitu *Snowball stemming library* yang didapat dari website snowball.tartarus.org. Proses dari *stemming* ditunjukkan pada Kode Program 5.6.

```

1  Public                               ArrayList<ArrayList<String>>
   stemming(ArrayList<ArrayList<String>> removestopword) {
2      ArrayList<ArrayList<String>>      stemming      =      new
   ArrayList<>();
3      englishStemmer stemmer = new englishStemmer();
4      ArrayList<ArrayList<String>>      stemmed      =      new
   ArrayList<>();
5
6      for (int i = 0; i < removestopword.size(); i++) {
7          stemming.add(new ArrayList<String>());
8          for (int j = 0; j < removestopword.get(i).size();
9  j++) {
10             stemming.get(i).add(j,
   removestopword.get(i).get(j));
11         }
12     }
13     for (int i = 0; i < stemming.size(); i++) {
14         stemmed.add(new ArrayList<String>());
15         for (int j = 0; j < stemming.get(i).size(); j++)
16     {
17         stemmer.setCurrent(stemming.get(i).get(j));
18         if (stemmer.stem()) {
19             stemmed.get(i).add(j,
   stemmer.getCurrent());
20         }
21     }
22     return stemmed;
23 }

```

Kode Program 5. 6 Implementasi *stemming*

Penjelasan Kode Program 5.6 adalah sebagai berikut:

1. Baris ke 3 dilakukan proses Inisialisasi penggunaan *stemmer* berbahasa Inggris.
2. Baris ke 6 ke 11 dilakukan pengaturan bahwa yang dilakukan *stemmer* yaitu pada *arraylist stemming*.
3. Baris ke 13 hingga 21 dilakukan proses *stemming* dan hasil dari proses *stemming* disimpan dalam *arraylist stemmed*.

5.1.7 TF-IDF

Proses TF-IDF merupakan salah satu proses untuk melakukan perhitungan bobot term pada suatu dokumen. Proses TF-IDF dilakukan dengan cara mengalikan frekuensi term pada dokumen atau disebut dengan TF dengan nilai *inverse* dari frekuensi dokumen yang mengandung term tersebut atau yang disebut sebagai IDF. Hasil dari TF-IDF ini akan digunakan dalam proses selanjutnya yaitu proses *cosine similarity*. Implementasi perhitungan nilai TF ditunjukkan pada Kode Program 5.7.

```

1 public String[][] pembobotanterm
  (ArrayList<ArrayList<String>> stemmed, ArrayList<String>
  term, int doc) {
2     String tf[][] = new String[term.size()][doc + 2];
3     double jmterm = 0;
4     int df = 0;
5     double idf = 0;
6
7     for (int i = 0; i < term.size(); i++) {
8         tf[i][0] = term.get(i);
9         for (int j = 0; j < tf[i].length - 2; j++) {
10            for (int k = 0; k < stemmed.get(j).size();
11            k++) {
12                if
13                (term.get(i).equalsIgnoreCase(stemmed.get(j).get(k))) {
14                    jmterm++;
15                }
16            }
17            if (jmterm == 0) {
18                jmterm = 0;
19            } else {
20                jmterm = 1 + Math.log10(jmterm);
21            }
22            tf[i][j] = String.valueOf(jmterm);
23            if (jmterm > 0) {
24                df++;
25            }
26            jmterm = 0;
27        }
28        idf = Math.log10((double) (doc) / df);
29        tf[i][tf[0].length - 1] = String.valueOf(idf);
30        tf[i][tf[0].length - 2] = String.valueOf(df);
31        df = 0;
32    }
33    return tf;
34 }
35
36 public String[][] tfidf(ArrayList<String> term, String[][]
37 tf, int doc) {
38     String tfidf[][] = new String[term.size()][doc];
39     for (int i = 0; i < term.size(); i++) {
40         tfidf[i][0] = term.get(i);
41         for (int j = 0; j < tfidf[0].length; j++) {
42             tfidf[i][j]
43             =
44             String.valueOf(Double.parseDouble(tf[i][j])
45             *
46             Double.parseDouble(tf[i][tf[0].length - 1]));
47         }
48     }
49 }

```

```

41     }
42     return tfidf;
43 }

```

Kode Program 5. 7 Implementasi TFIDF

Penjelasan Kode Program 5.7 adalah sebagai berikut:

1. Baris ke 11 hingga 12 merupakan proses untuk menghitung nilai term dari hasil *stemming*.
2. Baris ke 15 hingga 16 merupakan nilai log TF apabila jumlah term adalah 0 maka nilai log TF adalah 0.
3. Baris ke 17 hingga 18 menghitung apabila nilai TF bukan 0 maka akan menjumlahkan 1 dengan *log* dari nilai TF.
4. Baris 21 hingga 22 menghitung nilai DF.
5. Baris ke 26 menghitung nilai IDF dengan pembagian antara jumlah dokumen dengan nilai DF.
6. Baris ke 39 dilakukan proses perhitungan nilai TF-IDF yang didapatkan dari perkalian antara nilai *log* TF dengan nilai IDF.

5.1.8 Cosine Similarity

Tahap *Cosine similarity* merupakan tahap akhir dalam text similarity yang mana pada tahap cosine similarity ini akan menghasilkan nilai kesamaan antara dua dokumen. Proses perhitungan nilai *cosine similarity* ditunjukkan pada Kode Program 5.8.

```

1  public double[][] docq(String[][] tfidf, String[] tfidf_uji)
2  {
3      double[][] ti = new
4      double[tfidf.length][tfidf[0].length];
5      double[] ti_uji = new double[tfidf_uji.length];
6      double[][] hitungResult = new double[ti.length][];
7
8      for (int i = 0; i < ti.length; i++) {
9          for (int j = 0; j < ti[j].length; j++) {
10             ti[i][j] = Double.parseDouble(tfidf[i][j]);
11         }
12     }
13     for (int i = 0; i < ti_uji.length; i++) {
14         ti_uji[i] = Double.parseDouble(tfidf_uji[i]);
15     }
16     if (ti.length != ti_uji.length) {
17         throw new RuntimeException("Row number of First
18         Input and count of Second Input must be same.");
19     }
20     for (int i = 0; i < ti.length; i++) {
21         double[] currentValue = ti[i];
22         double[] hitungValue = new
23         double[currentValue.length];

```

```

24         for (int j = 0; j < currentValue.length; j++) {
25             hitungValue[j] = currentValue[j] * ti_uji[i];
26         }
27         hitungResult[i] = hitungValue;
28     }
29     return hitungResult;
30 }
31
32 public double[] hitungdocq(double[][] docq) {
33     double[][] transposed = transposeMatrix(docq);
34     double[] hasildocq = new double[transposed.length];
35     for (int i = 0; i < hasildocq.length; i++) {
36         double total = 0;
37         for (int j = 0; j < transposed[i].length; j++) {
38             total += transposed[i][j];
39         }
40         hasildocq[i] = total;
41     }
42     return hasildocq;
43 }
44
45 public double[] cosine(double[] hasildocq, double[] latih,
46 double[] uji, int doc) {
47     double[] cs = new double[doc];
48     for (int i = 0; i < cs.length; i++) {
49         cs[i] = hasildocq[i] / (Math.sqrt(latih[i]) *
50 Math.sqrt(uji[0]));
51     }
52     return cs;
53 }

```

Kode Program 5. 8 Implementasi *cosine similarity*

Penjelasan Kode Program 5.8 adalah sebagai berikut:

1. Baris ke 6 hingga 10 merupakan proses untuk mengubah dari string ke menjadi double ada data latih.
2. Baris ke 12 hingga 14 merupakan proses untuk mengubah dari string ke menjadi double pada data uji.
3. Baris ke 20 hingga 30 menghitung perkalian antara TFIDF data latih dengan TFIDF data uji.
4. Baris 32 hingga 43 menjumlahkan hasil dari perkalian TFIDF data latih dengan TFIDF data uji.
5. Baris ke 45 hingga 50 menghitung nilai cosine similarity.

BAB 6 HASIL DAN PEMBAHASAN

6.1 Hasil dan Analisis Prediksi Rating

Terdapat 31 *user* dan 90 sinopsis yang digunakan sebagai data latih untuk mendapatkan nilai prediksi rating pada tiap genre. *User* disini merupakan pengguna aktif goodreads.com yang memberikan rating pada beberapa sinopsis. Namun terdapat *user* yang tidak memberikan rating pada semua sinopsis yang mana data tersebut terdapat pada Lampiran 1. *Genre based collaborative filtering* digunakan sebagai penghitung prediksi rating yang diberikan *user* pada masing-masing sinopsis. Tabel 6.1 menunjukkan hasil dari prediksi rating yang diberikan oleh *user* pada masing-masing sinopsis dengan menggunakan *genre based collaborative filtering*.

Tabel 6. 1 Hasil prediksi rating oleh user untuk masing-masing sinopsis

Sinopsis	User							
	User1	User2	User3	User4	User5	...	User 30	User31
1	4	4	3	4	5	...	4	4
2	4	3	5	5	5	...	4	4
3	5	4	4	5	5	...	4	5
4	5	4	5	5	5	...	4	4
5	4	4	3	4	5	...	4	4
...
88	3	4	5	4	4	...	1	1
89	4	4	5	5	4	...	1	5
90	1	2	4	4	4	...	1	4

Setelah diketahui semua rating yang diberikan oleh *user* ke masing-masing sinopsis, selanjutnya mencari rata-rata rating berdasarkan genre sinopsis tersebut. Terdapat tujuh genre diantaranya yaitu *fiction*, *fantasy*, *horror*, *sci-fic*, *mystery*, *thriller*, dan *romance*. Rata-rata rating berdasarkan genre ini yang dijadikan sebagai prediksi rating berdasarkan genrenya. Hasil dari rata-rata rating ditunjukkan pada Tabel 6.2.

Tabel 6. 2 Hasil rata-rata rating per genre

Genre	Nomor sinopsis per genre	Rata-rata rating genre
<i>Fiction</i>	1 - 18	3,93
<i>Fantasy</i>	19 - 34	3,88
<i>Horror</i>	35 - 45	3,98

Genre	Nomor sinopsis per genre	Rata-rata rating genre
<i>Sci-fic</i>	46 - 57	3,85
<i>Mystery</i>	58 - 70	3,86
<i>Thriller</i>	71 - 82	3,72
<i>Romance</i>	83 - 90	3,56

Setelah diketahui hasil dari perhitungan prediksi rating berdasarkan genre dengan menggunakan genre based collaborative filtering, dilakukan pengujian pada sistem dengan menggunakan data uji. Terdapat 35 sinopsis dari 7 genre yang berbeda dengan masing-masing genre terdapat 5 sinopsis yang dijadikan sebagai data uji. Hasil dari pengujian data uji terhadap sistem ditunjukkan pada Tabel 6.3.

Tabel 6. 3 Hasil pengujian sistem

No	Genre Asli	Rating Asli	Klasifikasi Genre	Rating Prediksi
1	Fiction	4,08	Fiction	3,93
2	Fiction	3,94	Sci-fic	3,85
3	Fiction	3,7	Romance	3,56
4	Fiction	4,4	Ficiton	3,93
5	Fiction	4,21	Fiction	3,93
6	Fantasy	4,09	Fantasy	3,88
7	Fantasy	4,07	Sci-fic	3,85
8	Fantasy	4,09	Fantasy	3,88
9	Fantasy	3,99	Sci-fic	3,85
10	Fantasy	4,47	Fantasy	3,88
11	Horror	4,18	Mystery	3,86
12	Horror	3,77	Sci-fic	3,85
13	Horror	4,15	Thriller	3,72
14	Horror	3,39	Horror	3,98
15	Horror	4,32	Sci-fic	3,85
16	Sci-fic	3,71	Fantasy	3,88
17	Sci-fic	3,68	Sci-fic	3,85
18	Sci-fic	4,03	Mystery	3,86
19	Sci-fic	4,15	Sci-fic	3,85
20	Sci-fic	4,51	Sci-fic	3,85
21	Mystery	3,97	Thriller	3,72

No	Genre Asli	Rating Asli	Klasifikasi Genre	Rating Prediksi
22	Mystery	3,74	Sci-fic	3,85
23	Mystery	3,85	Mystery	3,86
24	Mystery	4,34	Horror	3,98
25	Mystery	3,88	Mystery	3,86
26	Thriller	3,86	Thriller	3,72
27	Thriller	3,87	Romance	3,56
28	Thriller	4,15	Mystery	3,86
29	Thriller	3,78	Mystery	3,86
30	Thriller	3,85	Thriller	3,72
31	Romance	4,02	Thriller	3,72
32	Romance	3,93	Fiction	3,93
33	Romance	4,35	Romance	3,56
34	Romance	3,51	Fantasy	3,88
35	Romance	4,22	Romance	3,56

Dari hasil pengujian sistem terhadap prediksi rating berdasarkan genre, dapat dihitung nilai MAE dengan cara mengurangi nilai hasil prediksi dengan nilai yang sebenarnya, selanjutnya dibagi dengan jumlah keseluruhan data latih seperti pada persamaan 2.9.

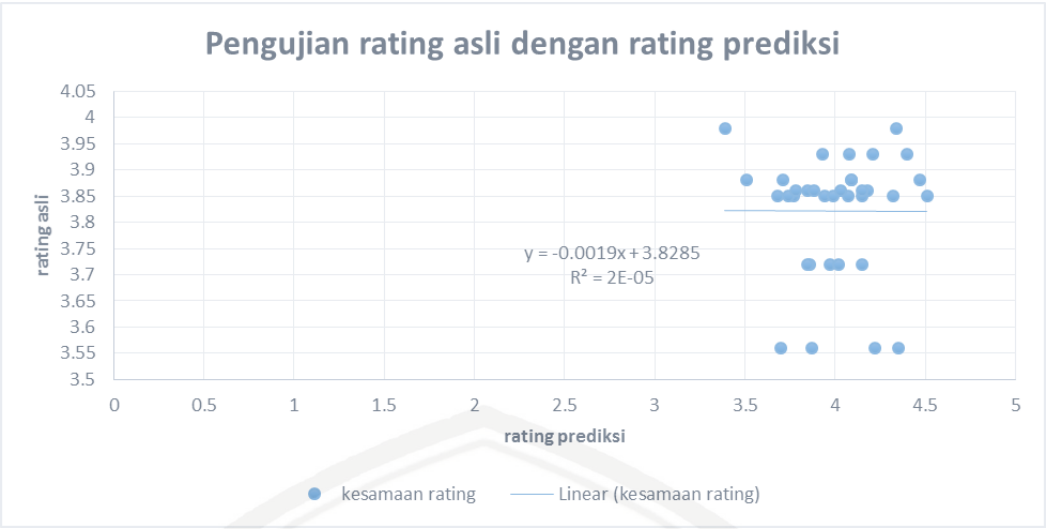
$$MAE = \frac{|3.93 - 4.08| + |3.85 - 3.94| + \dots + |3.88 - 3.51| + |3.56 - 4.22|}{35}$$

$$MAE = \frac{|-0.15| + |-0.09| + \dots + |0.37| + |-0.66|}{35}$$

$$MAE = \frac{9.71}{35}$$

$$MAE = 0,27742857$$

Dari perhitungan Mean Absolute Error, nilai yang dihasilkan MAE termasuk kecil yaitu sebesar 0,27742857. Hal ini dikarenakan nilai rating yang sesungguhnya dengan nilai rating prediksi yang tidak terlalu jauh berbeda. Kemudian dilihat dari diagram *scatter* dari hasil pengujian rating pada data uji terhadap sistem diperoleh bahwa nilai rating dapat diprediksi sebesar -0,0019 kali nilai rating asli ditambah dengan nilai konstan sebesar 3,8285 dengan nilai koefisien korelasi sebesar 0,00002. Grafik pengujian rating asli dengan rating prediksi ditunjukkan pada Gambar 6.1.



Gambar 6. 1 Grafik pengujian rating asli dengan rating prediksi

6.2 Hasil dan Analisis Klasifikasi Sinopsis

Berdasarkan dari hasil pengujian genre data uji terhadap sistem diperoleh bahwa jumlah data latih tidak memberikan pengaruh yang besar pada hasil prediksi untuk genre. Analisis dari pengujian data uji terhadap sistem ditunjukkan pada Tabel 6.3.

Tabel 6. 4 Hasil analisis pengujian

Data latih			Data yang dihasilkan		
Genre	Data	Jumlah term	Kesamaan Genre	Jumlah term	Term
Ficiton	18	985	Ficiton (3), Romance (1)	32	Country (1), people (2), citi (1), explod (1), violenc (1), charact (1), profound (1), tell (1), unforgett (1), stori (2), time (2), beartown (4), community (1), town (1), hockey (2), team (4), match (1), resid (1), bring (1), mile (1), start (1), world (2), high (1), life (2), meet (1), live (2), experi (1), heart (1), begin (1), musician (1), drawn (1), great (1), music (2), back (1), long (1)
Fantasy	16	573	Fantasy (3), Sci-fic (1),	25	Pass (1), unearth (1), lurk (1), depth (1), narrow (1), beneath (1), sweet (1),



Data latih			Data yang dihasilkan		
Genre	Data	Jumlah term	Kesamaan Genre	Jumlah term	Term
			Romance(1)		merci (1), blood (1), hone (1), novic (1), blade (1), fist (1), nona (3), grey (1), enemi (1), empir (1), assassin (1), protect (2), trust (1), guard (1), head (2), adventur (1), guest (1), profession (1)
Horror	11	289	Horror (1), mystery (1)	5	Stranger (2), quiet (1), privat (2), strike (8), busi (1)
Sci-fic	12	292	Fiction (1), Fantasy (2), Sci-fic (3), Horror (2), Mystery (1)	14	Franklin (3), metal (1), piec (1), defens (1), corp (1), knowledge (1), problem (1), thrive (1), planet (2), chao (2), faster (1), river (1), soldier (1), possibl (1)
Mystery	13	265	Horror (1), Sci-fic (1), Mystery (2), Thriller (2)	8	spent (1), bone (1), funer (1), aaron (1), falk (3), reluct (1), luke (5), afraid (1)
Thriller	12	173	Horror (1), Mystery (1), Thriller (2), Romance(1)	4	alarm (1), wine (1), window (1), lesson (1)
Romance	8	190	Fiction (1), Thriller (1), Romance(2)	4	resort (3), arrog (1), push (1), fill (1)

Dari hasil klasifikasi sinopsis dapat diketahui nilai akurasi yang didapatkan dari tingkat kedekatan antara nilai prediksi dengan nilai aktual. Dari hasil pengujian

yang telah dilakukan, dapat diketahui hasil akurasi dengan menggunakan persamaan 2.8 sebagai berikut:

$$Accuracy = \frac{16}{35} \times 100\%$$

$$Accuracy = 45,714286 \%$$

Akurasi yang dihasilkan dari klasifikasi dengan text similarity hanya menghasilkan nilai akurasi sebanyak 45,714286 %. Tabel 6.4 menunjukkan bahwa genre *fiction* yang memiliki data latih yang paling banyak yaitu sebanyak 18 dengan jumlah term sebanyak 985 hanya menghasilkan 4 genre yang sama yaitu *fiction* yang memiliki data latih sebanyak delapan belas sinopsis hanya menghasilkan prediksi genre *fiction* sebanyak empat dari tiga puluh lima data uji yang mana tiga diantaranya merupakan genre *fiction* asli sedangkan satu yang lainnya merupakan dari genre *romance*. Sedangkan genre *fantasy* yang memiliki data latih sebanyak 16 dengan jumlah term 573 menghasilkan kesamaan genre sebanyak lima dari tiga puluh lima data uji yang mana tiga diantaranya berasal dari genre *fantasy*, satu dari genre *sci-fic*, dan satu dari genre *romance*. Selanjutnya genre *horror* yang memiliki 298 term atau data latih sebanyak sebelas hanya menghasilkan kesamaan genre dua yang mana satu diantaranya dari genre *horror* dan satu lainnya dari genre *mystery*. Sedangkan genre *sci-fic* yang hanya memiliki data latih sebanyak dua belas dengan jumlah term sebanyak 292 menghasilkan kesamaan genre paling banyak yaitu sejumlah Sembilan. Satu berasal dari genre *fiction*, dua dari genre *fantasy*, tiga dari genre *sci-fic*, dua dari genre *mystery*, dua dari genre *horror*, dan satu dari genre *mystery*. Kemudian untuk genre *mystery* yang memiliki tiga belas data latih menghasilkan enam kesamaan genre yang sama diantaranya satu dari genre *horror*, satu dari genre *sci-fic*, dua dari genre *mystery*, dan dua dari genre *thriller*. Untuk genre *thriller* sendiri yang memiliki term pada data latih sebanyak 173 menghasilkan lima kesamaan genre yang mana dua diantaranya merupakan dari genre *thriller* sendiri, satu dari genre *horror*, satu dari genre *mystery*, dan satu lainnya dari genre *romance*. Sedangkan genre *romance* yang memiliki data latih paling sedikit yaitu delapan data latih menghasilkan kesamaan genre sebanyak empat, diantaranya dari genre *fiction* sebanyak satu, dari genre *thriller* sebanyak satu, dan dua dari genre *romance* itu sendiri.

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil dari pengujian dan analisis mengenai prediksi rating novel baru dengan menggunakan *text similarity* dan *genre based collaborative filtering* maka diperoleh kesimpulan sebagai berikut:

1. Dalam perancangan prediksi rating novel baru menggunakan *genre based collaborative filtering* dan *text similarity* diperlukan beberapa proses tahapan yaitu yang pertama merupakan proses prediksi rating yang masih kosong dengan menggunakan *genre based collaborative filtering*. Selanjutnya menggunakan *text similarity* dimana di dalam *text similarity* terdapat proses *preprocessing*, pembobotan dengan menggunakan TFIDF, serta perhitungan kemiripan dokumen uji dengan dokumen latih dengan menggunakan *cosine similarity*. Dalam proses *preprocessing* terdapat lima proses yaitu *tokenization*, *cleaning*, *case folding*, *filtering*, serta *stemming*.
2. Salah satu keunggulan dari sistem ini yaitu jumlah data latih yang tidak tertalu mempengaruhi hasil dari prediksi kedekatan nilai *cosine similarity*. Seperti pada genre *ficiton* yang memiliki data latih sebanyak delapan belas sinopsis hanya menghasilkan prediksi genre *fiction* sebanyak empat dari tiga puluh lima data uji yang mana tiga diantaranya merupakan genre *fiction* asli sedangkan satu yang lainnya merupakan dari genre *romance*. Hal tersebut mempengaruhi akurasi sehingga akurasi yang dihasilkan dari *text similarity* yaitu sebesar 45,714286 % . Sedangkan kekurangan dari sistem ini yaitu *user* dipaksa untuk memiliki keminatan pada semua genre dan memberikan rating pada semua jenis genre sehingga menyebabkan nilai rating yang hampir sama antara nilai yang sesungguhnya dengan nilai prediksi.

7.2 Saran

Dari penelitian yang telah dilakukan terdapat beberapa saran yang dapat dijadikan acuan untuk penelitian selanjutnya, diantaranya yaitu:

1. Fitur yang digunakan dapat hanya berupa rating rata-rata novel dan sinopsis.
2. Dapat menggunakan metode lain untuk melakukan klasifikasi seperti *Support Vector Machine* (SVM) karena metode ini memiliki tingkat keakuratan yang baik.

DAFTAR REFERENSI

- Advertising-Indonesia, 2017. *Mengenal Manfaat dan Kekurangan Rating*. [Online] Available at: <http://advertising-indonesia.id/2017/09/18/mengenal-manfaat-dan-kekurangan-rating/> [Diakses 15 February 2018].
- Amazon.com, 2018. *Top 20 lists in Books*. [Online] Available at: <https://www.amazon.com/> [Diakses 23 November 2018].
- Devasena, C. L., 2013. Classification of Multivariate Data Sets Without Missing Values Using Memory Based Classifiers - An Effectiveness Evaluation. *International Journal of Artificial Intelligence & Applications (IJAI)*, Volume 4 No. 1, hal. 129-142.
- Goodreads, 2018. *goodreads.com*. [Online] Available at: <https://www.goodreads.com/about/us> [Diakses 19 11 2018].
- Harper, J., 2016. *The Dry*. [Online] Available at: <https://www.goodreads.com/book/show/27824826-the-dry> [Diakses 19 November 2018].
- Herwijayanti, B., Ratnawati, D. E. & Muflikhah, L., 2018. Klasifikasi Berita Online dengan menggunakan Pembobotan TF-IDF dan Cosine Similarity. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Volume 2, hal. 306-312.
- ISBN, 2014. *Traditional Print Book Production Dipped Slightly in 2013*. [Online] Available at: <http://www.bowker.com/news/2014/Traditional-Print-Book-Production-Dipped-Slightly-in-2013.html> [Diakses 13 February 2018].
- Jane, 2014. *On the importance of pseudonymous activity*. [Online] Available at: <https://dearauthor.com/features/essays/on-the-importance-of-pseudonymous-activity/> [Diakses 9 11 2018].
- Kesuma, H. W. A. & Pribadi, F. S., 2016. Penerapan Cosine Similarity dalam Aplikasi Kitab Undang-Undang Hukum Dagang (Wetboek Van Koophandle Voor Indonesia). *Jurnal Teknik Elektro*, Volume Vol. 8 No. 1, hal. 18-20.
- King, S. & King, O., 2017. *Sleeping Beauties*. [Online] Available at: <https://www.goodreads.com/book/show/344466922-sleeping-beauties> [Diakses 19 November 2018].
- Kurniawan, A., Solihin, F. & Hastarita, F., 2014. Perancangan dan Pembuatan Aplikasi Pencarian Informasi Beasiswa dengan Menggunakan Cosine Similarity. *Jurnal SimanteC*, Volume 4, hal. 115-124.



- Li, N. & Zhang, P., 2002. *Consumer Online Shopping Attitudes and Behavior: An Assessment of Research*. Dallas, AMCIS.
- Mustafa, N., Ibrahim, A. O., Ahmed, A. & Abdullah, A., 2017. *Collaborative Filtering: Techniques and Applications*. Khartoum, s.n.
- Ng, C., 2017. *Little Fires Everywhere*. [Online] Available at: <https://www.goodreads.com/book/show/34273236-little-fires-everywhere> [Diakses 19 November 2018].
- Nurgiyantoro, B., 1995. *Teori Pengkaji Fiksi*. Yogyakarta: Gadjah Mada University Press.
- Pardiyono, 2007. *Pasti Bisa!: Teaching Genre-Based Writing*. Yogyakarta: Andi Offset.
- Puntheeranurak, S. & Chaiwitooanukool, T., 2011. An Item-based Collaborative Filtering Method using Item-based Hybrid Similarity. hal. 469-472.
- Spaeth, A. & Desmarais, M. C., 2013. *Combining Collaborative Filtering and Text Similarity for Expert Profile Recommendations in Social Websites*. Canada, s.n.
- Sulistyo, M. E., Saptono, R. & Asshidiq, A., 2015. Penilaian Ujian Bertipe Essay Menggunakan Metode Text Similarity. *Telematika*, Volume 12, hal. 146-158.
- Tarigan, H. G., 2011. *Pengajaran Analisis Kesalahan Berbahasa*. Bandung: Angkasa.
- Thelwall, M. & Kousha, K., 2016. Goodreads: A Social Network Site for Book Readers. *Journal of the Association for Information Science and Technology*, hal. 1-12.
- Utami, L. D. & Wahono, R. S., 2015. Integrasi Metode Information Gain Untuk Seleksi Fitur dan Adaboost Untuk Mengurangi Bias Pada Analisis Sentimen Review Restoran Menggunakan Algoritma Naive Bayes. *Journal of Intelligent Systems*, Volume 1, hal. 120-126.
- Wahyuni, R. T., Prastiyanto, D. & Supraptono, E., 2017. Penerapan Algoritma Cosine Similarity dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi. *Jurnal Teknik Elektro*, Volume 9, hal. 18-23.
- Wang, M.-j. & Han, J.-t., 2012. Collaborative Filtering Recommendation Based on Item Rating and Characteristic Information Prediction. hal. 214-217.
- Weir, A., 2017. *Artemis*. [Online] Available at: <https://www.goodreads.com/book/show/34928122-artemis> [Diakses 19 November 2018].
- Zhang, Y. & Song, W., 2009. *A Collaborative Filtering Recommendation Algorithm Based on Item Genre and Rating Similarity*. Jinzhou, s.n., hal. 72-75.