# SELEKSI FITUR INFORMATION GAIN UNTUK KLASIFIKASI INFORMASI TEMPAT TINGGAL DI KOTA MALANG BERDASARKAN TWEET MENGGUNAKAN METODE NAÏVE BAYES DAN PEMBOBOTAN TF-IDF-CF

**SKRIPSI**

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Ahmad Efriza Irsad
NIM: 155150201111369


PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2019

# PENGESAHAN

**SELEKSI FITUR INFOROMATION GAIN UNTUK KLASIFIKASI INFORMASI TEMPAT TINGGAL DI KOTA MALANG BERDASARKAN TWEET MENGGUNAKAN METODE NAÏVE BAYES DAN PEMBOBOTAN TF-IDF-CF**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Ahmad Efriza Irsad
NIM: 155150201111369

Skripsi ini telah diuji dan dinyatakan lulus pada
21 Mei 2019
Telah diperiksa dan disetujui oleh:

| Pembimbing I | Pembimbing II |
|---|---|
| Yuita Arum Sari, S.Kom., M.Kom. | M. Ali Fauzi, S. Kom., M.Kom. |
| NIK: 201609 880715 2 001 | NIK: 201502 890101 1 001 |

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

Malang, 21 Mei 2019


Ahmad Efriza Irsad
NIM: 155150201111369

# PRAKATA

Dengan menyebut nama Allah SWT yang Maha Pengasih lagi Maha Penyayang. Puji syukur atas kehadirat-Nya yang telah melimpahkan Rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul: "Seleksi Fitur Information Gain Untuk Klasifikasi Informasi Tempat Tinggal Di Kota Malang Berdasarkan Tweet Menggunakan Metode Naïve Bayes Dan Pembobotan TF-IDF-CF".

Skripsi ini merupakan salah satu syarat kelulusan yang harus ditempuh di Fakultas Ilmu Komputer, Program Studi Teknik Informatika Universitas Brawijaya Malang. Penulis menyadari, skripsi ini tidak akan berhasil tanpa bantuan dan dukungan banyak pihak, pada kesempatan ini penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Yuita Arum Sari, S. Kom., M.Kom., selaku dosen pembimbing 1 dan M. Ali Fauzi, S. Kom., M.Kom., selaku dosen pembimbing 2 yang telah memberikan bimbingan, saran, serta arahan selama penyusunan 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 Tri Astoto Kurniawan, S.T., M.T., Ph.D selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya Malang.
4. Seluruh dosen Fakultas Ilmu Komputer yang telah mendidik dan memberikan ilmu serta wawasannya selama perkuliahan dan civitas akademika Fakultas Ilmu Komputer yang telah memberikan dukungan dan bantuan selama menempuh perkuliahan dan menyelesaikan skripsi ini.
5. Kedua Orang Tua, kakak, dan seluruh keluarga besar atas segala doa, dukungan, nasihat, dan kasih sayang yang senantiasa diberikan kepada penulis demi terselesaikannya skripsi ini.
6. Teman-teman Kontrakan yaitu, Ivan, Juan, Prim, Stefan, Rahman, dan Ibnu.

Penulis menyadari bahwa banyak kekurangan dalam skripsi ini baik dalam penulisan dan isi bahasannya. Oleh karena itu penulis menerima segala saran dan kritik untuk menyempurnakan skripsi ini. Akhir kata penulis berharap semoga skripsi ini dapat memberikan manfaat maupun inspirasi terhadap semua pihak.

Malang, 21 Mei 2019

Penulis
efriza88@gmail.com

# ABSTRAK

**Ahmad Efriza Irsad, Seleksi Fitur Information Gain Untuk Klasifikasi Informasi Tempat Di Kota Malang Berdasarkan Tweet Menggunakan Metode Naïve Bayes Dan Pembobotan TF-IDF-CF**

**Pembimbing: Yuita Arum Sari S.Kom., M.Kom dan M. Ali Fauzi, S.Kom, M.Kom.,**

Kota malang merupakan kota yang memiliki peningkatan jumlah penduduk yang bisa dibilang cukup pesat, yaitu sekitar 50 ribu jiwa dalam jangka waktu 5 tahun. Salah satu penyebabnya dikarenakan kota Malang merupakan salah satu kota pendidikan karena di kota ini terdapat banyak perguruan tinggi yang cukup banyak dan bisa dibilang cukup populer, seperti Universitas Brawijaya (UB), Universitas Islam Malang (Unisma), dll. Hal ini membuat banyak pendatang dari luar daerah kota Malang berkuliah di kota malang, ada beberapa hal yang mungkin menjadi alasan pendatang memilih kota Malang, salah satunya karena kota Malang memiliki universitas dengan kualitas yang bisa dibilang salah satu yang terbaik di Inonesia. Ketika menjadi seorang migran tentu yang dibutuhkan adalah tempat tinggal dalam jangka waktu yang panjang, karena itu para pendatang tadi tentu memerlukan informasi tempat tinggal berupa kost atau kontrakan untuk ditinggali, informasi tentang tempat tinggal ini dapat kita dapatkan melalui media sosial seperti Twitter, namun di Twitter masih belum ada pengelompokkan mengenai informasi-informasi seperti ini. Melihat masalah ini maka digunakan Teknik klasifikasi untuk mengelompokkan informasi berupa tempat tinggal yang ada di kota malang. Pada penelitian ini digunakan metode *Naïve Bayes* sebagai metode pengklasifikasian dan metode *Information Gain* untuk menyeleksi fitur yang digunakan. Sebelum masuk kedalam proses pengklasifikasian dilakukan pembobotan terlebih dahulu menggunakan metode TF-IDF-CF. Data yang digunakan sebagai data latih sebanyak 150 data, sedangkan 60 data untuk data uji. Hasil akurasi terbaik yang didapatkan dari penelitian ini adalah sebesar 71,66% dengan menggunakan fitur sebanyak 33%, pembobotan TF-IDF-CF, dan tanpa penggunaan fitur angka.

**Kata kunci**: kost, kontrakkan*,* Twitter*, Naïve Bayes, Information Gain,* TF-IDF-CF

# ABSTRACT

**Ahmad Efriza Irsad,** ***Information Gain as Feature Selection for Classification Of Residence Information in Malang City Based On Tweet Using Naïve Bayes Method and TF-IDF-CF Weighting***

***Supervisors*: Yuita Arum Sari S.Kom., M.Kom and M. Ali Fauzi, S.Kom, M.Kom.,**

*Malang city is a city that has a significant increase in population, which is around 50 thousands people in just period of 5 years. One of the reasons is because Malang city is a city of education, the reasons its called city of education is because in this City there are a lot of public university and private university that are quite popular, such as Universitas Brawijaya (UB), Universitas Islam Malang (Unisma), etc. This resulted many migrants from outside the area of Malang city study in Malang city. There are some things that might be the reasons why migrants choose Malang city, such as the Malang city have one of the best quality university in Indonesia. When becoming a migrant, the most needed thing is certainly a place to live in a long term, because of that the migrants need information on where to live in the form of boarding house or rent house to live in, we can get this kind of information trough social media like Twitter, but on Twitter there is still no category for this kind of information. By seeing this problem, we can use Classification technique to classified the information in the form of living quarters in the city of Malang. In this study Naïve Bayes method is used as the classification method, and Information gain as the feature selection method. Before entering the classification process the weighting is done first using TF-IDF-CF method. This study uses 150 training data and 60 testing data. The highest accuracy value in this study are 71,66% using 33% of feature, using TF-IDF-ICF weighthing and, without using number feature.*

***Keywords***: *boarding house, rent house, Naïve Bayes, Information Gain, TF-IDF-CF*

# DAFTAR ISI

# DAFTAR GAMBAR

# DAFTAR TABEL

# BAB 1 PENDAHULUAN

## 1.1 Latar belakang

Kota Malang merupakan kota yang mengalami peningkatan jumlah penduduk yang bisa dibilang cukup pesat data dari Dinas Kependudukan dan Catatan Sipil (Dispendukcapil), diperkirakan dari tahun 2012 hingga 2017 penduduk di kota malang mengalami kenaikan dari total 845.271 menjadi 895.387, yang tentunya jumlah ini bukan jumlah yang sedikit. Pertambahan penduduk ini tak lepas dari warga dari luar Kota Malang yang pindah ke kota Malang atau bisa disebut migrasi, migrasi ini menjadi salah satu faktor signifikan pertambahan penduduk di kota Malang, penyebab pendatang pindah ke kota Malang sendiri ada beberapa faktor, karena kota Malang adalah perkotaan, pusat perekonomian, dan pusat Pendidikan (Abidin, 2017).

Kota Malang dikenal sebagai salah satu kota Pendidikan di Indonesia, dimana berdasarkan data dari TIM MalangTODAY, setidaknya terdapat 7 perguruan tinggi negeri (PTN) dan perguruan tinggi swasta (PTS) yang menjadi tujuan mahasiswa baru di kota Malang, perguruan tinggi tersebut antara lain, Universitas Brawijaya (UB), Universitas Negeri Malang (UM), Politeknik Negeri Malang (Polinema), Institut Teknologi Nasional (ITN), Universitas Merdeka (Unmer), Universitas Islam Malang (Unisma), dan Universitas Widyagama, dimana pada tahun 2018 saja terdapat sekitar 22.687 mahasiswa baru yang tersebar di perguruan tinggi yang telah disebutkan di atas.

Mahasiswa yang berada di kota Malang sendiri tidak sedikit merupakan migran, atau orang yang melakukan migrasi. Hal ini dikarenakan beberapa hal, seperti contohnya mungkin saja daerah asal migran tadi memiliki perguruan tinggi yang kualitasnya masih kalah dengan yang berada di kota Malang, sehingga mereka mengikuti proses seleksi masuk universitas dengan skala nasional. Dengan ini menyebabkan pada mahasiswa tadi perlu memenuhi kebutuhannya secara mandiri, entah itu kebutuhan primer, sekunder, atau tersier. Sebagai manusia kebutuhan sandang, pangan, dan, papan merupakan kebutuhan primer, tak terkecuali bagi pendatang, namun hal yang bisa dibilang paling utama bagi pendatang terutama yang bermigrasi keluar kota adalah tempat tinggal, di kota Malang sendiri tak sedikit pendatang adalah mahasiswa, tentu tempat tinggal seperti kos dan kontrakkan sangat diperlukan untuk keperluan tempat tinggal, untuk mendapatkan tempat tinggal ini sendiri pendatang tentu memerlukan informasi tentang ketersediaan kos atau kontrakan untuk ditinggali. Cukup banyak informasi tentang kontrakkan dan kos pada sosial media seperti Instagram, Twitter, dan yang lainnya, meski bisa dibilang informasi tersebut masih belum terkelompokkan dan kadang masih sulit dicari.

Twitter merupakan sosial media yang saat ini bisa dibilang cukup banyak menyediakan informasi bahkan informasi yang bisa dibilang *real-time*, namun pada Twitter sendiri bisa dibilang *tweet-tweet* yang ada masih memiliki cakupan yang sangat luas, sehingga diperlukan klasifikasi untuk memudahkan pencarian

informasi yang diinginkan. Klasifikasi teks ini sendiri memiliki beberapa metode seperti, *support vector machine* (SVM), *artifial neural network*, dan *probabilistic.* Pada penelitian sebelumnya terdapat sebuah penelitian yang membandingkan membandingkan metode klasifikasi teks *Naïve Bayes Classifier* (NBC) dengan metode SVM, C4.5, dan *K-Nearest Neighbor* (K-NN), dimana hasil penelitian tersebut menunjukkan akurasi masing-masing metode adalah SVM akurasi 92%, NBC akurasi 90%, C4.5 Akurasi 77,5%, dan K-NN akurasi 50% (Wulandini dan Nugroho, 2009). Dapat kita lihat metode untuk klasifikasi dengan nilai akurasi tertinggi adalah NBC dan SVM, yang bisa dibilang menghasilkan akurasi yang cukup signifikan dibandingkan dengan metode lain.

Pada pengklasifikasian dokumen ini memiliki permasalahan yaitu ketika terdapat kata-kata yang tidak relevan atau bisa disebut sebagai *noise*, banyaknya kata yang mampu merepresentasikan sebuah dokumen terkadang menjadi sebuah permasalahan pada proses pengklasifikasian teks karena tingginya dimensi fitur yang ada, fitur ini mampu mencapai jumlah hingga ribuan hingga ratusan ribu fitur unik, hal ini akan berakibat pada kinerja klasifikasi (Putra, Sudarma, dan Kumara, 2016). Untuk mencegah hal ini terjadi dilakukan proses seleksi fitur yang berfungsi untuk memilih fitur-fitur yang paling relevan dalam merepresentasikan sebuah dokumen pada kategori tertentu, pada penelitian yang dilakukan oleh George Forman (Forman, 2003), terdapat beberapa metode seleksi fitur untuk klasifikasi teks, seperti *document frequency* (DF), *mutual information* (MI), *information gain* (IG), $x^2$ *statistic* (CHI), dan *term strength* (TS). Pada penelitian ini metode seleksi fitur yang digunakan adalah IG karena IG merupakan metode seleksi fitur yang sederhana dan efisien dalam mengukur dan mengetahui ada tidaknya fitur pada dokumen kemudian memilih subset optimal (Putra, Sudarma, dan Kumara, 2016), dan menurut penelitian Smita Chormunge dan Sudarson Jena (Chormunge dan Jena, 2016) IG adalah metode paling sederhana untuk memeringkat atribut, metode ini banyak digunakan untuk klasifikasi teks. Metode IG mampu memilih fitur yang paling merepresentasikan sebuah dokumen dan juga mampu meningkatkan performa klasifikasi (Chormunge dan Jena, 2016).

Sebelum pemrosesan klasifikasi diperlukan pembobotan fitur untuk membantu meningkatkan akurasi pada saat klasifikasi, pembobotan yang umum digunakan adalah *Term frequency – Inverse Document Frequency* (TD-IDF)*,* TF-IDF hanya mempertimbangkan parameter frekuensi kemunculan fitur dalam dokumen dan jumlah dokumen yang mengandung fitur tersebut. Padahal, dalam proses klasifikasi terdapat informasi lain seperti, frekuensi kemunculan fitur dalam setiap kelas, distribusi kemunculan fitur, dalam setiap kelas, dan jumlah dokumen pada setiap kelas, karena itu pada TF-IDF ini ditambahkan parameter *class frequency*, yang berguna untuk menghitung *term frequency* pada suatu kelas, metode ini diberi nama TF-IDF-CF (Liu dan Yang, 2012).

Berdasarkan dari penjelasan di atas pada penelitian ini digunakan metode NBC dalam pengklasifikasian teks dari Twitter, karena dapat dilihat pada pemaparan di atas dari seluruh metode yang dibandingkan NBC dan SVM memiliki hasil dengan akurasi tertinggi yang bisa dibilang cukup signifikan dbanding metode

klasifikasi lainnya, meskipun SVM memiliki hasil klasifikasi tertinggi,pemilihan metode NBC didasari karena NBC memiliki kesederhanaan dan mampu memberikan performa yang sangat baik dalam proses pengklasifikasian (Lewis,1998), pada beberapa penelitian sebelumnya, seperti penelitian untuk klasifikasi lirik menggunakan Naïve Bayes, mampu memberikan hasil evaluasi yang mampu mencapai presisi hingga 0,93, *recall* sebesar 0,95, dan *F-measure* sebesar 0,94, tentu hasil tersebut mampu membuktikan bahwa *Naïve Bayes* merupakan model klasifikasi yang sangat baik (Bužić dan Dobša, 2018), kemudian ada penelitian tentang *Hate Speech* berbahasa Indonesia dengan dataset dari Twitter mampu memperlihatkan hasil pengujian dengan menggunakan *Naïve Bayes* mencapai akurasi hingga 93 % (Fatahillah, Suryati, dan Haryawan, 2018), dengan ditambahkan seleksi fitur dengan metode *information gain* yang berguna untuk memilih fitur yang lebih merepresentasikan kelas untuk klasifikasi dan pembobotan TF-IDF yang dikembangkan menjadi TF-IDF-CF.

Dengan adanya pembobotan TF-IDF-CF model pengklasifikasian *naïve bayes* yang digunakan adalah gaussian karena gaussian adalah model pengklasifikasian naïve bayes yang menggunakan tipe data kontinu (Capdevila dan Flórez 2009), dengan adanya pembobotan tersebuat diharapkan mampu meningkatkan akurasi pengklasifikasian, sistem yang dibuat diharapkan mampu mengkategorikan *tweet* berdasarkan kecenderungan kata yang berada didalam setiap kategorinya. Tujuan utama dari penelitian ini adalah penelitian ini diharapkan dapat mempermudah pendatang atau perantau untuk mendapatkan informasi yang spesifik tentang tempat tinggal yang berada di kota Malang khususnya kos dan kontrakkan.

Pada penelitian ini diperlukan pengklasifikasian dikarenakan meski dari pemilihan kategori yang digunakan terlihat bahwa tanpa klasifikasi saja data tersebut dapat dikelompokkan dengan menggunakan rule-based, namun pada penelitian ini data yang digunakan adalah data yang didapatkan melalui Twitter, dimana data dari Twitter ini memiliki data berbahasa Indonesia yang tidak baku sehingga terdapat kata-kata yang seharusnya menjadi salah kata yang merepresentasikan sebuah kelas tidak menjadi kata yang mampu merepresentasikan kelas tersebut. Sehingga diperlukan klasifikasi untuk mengetahui kecenderungan antar kategorinya.

## 1.2 Rumusan masalah

Dari latar belakang yang telah dijelaskan sebelumnya, maka telah ditentukan rumusan masalah untuk penelitian adalah sebagai berikut ini:

1. Bagaimana hasil akurasi yang didapatkan dari klasifikasi informasi di kota Malang yang sudah di implementasikan?
2. Bagaimana pengaruh pembobotan TF-IDF-CF seleksi fitur *Information Gain* pada metode NBC untuk klasifikasi informasi di kota malang

## 1.3 Tujuan

Setelah rumusan masalah ditentukan, maka selanjutnya ditentukan juga tujuan akhir dari penelitian yang akan dilakukan adalah sebagai berikut:

1. Mengetahui hasil akurasi sistem klasifikasi yang telah diterapkan
2. Mengetahui bagaimana pengaruh pembobotan TF-IDF-CF dan seleksi fitur *Information Gain* pada metode *Naïve Bayes* untuk klasifikasi informasi tempat di kota Malang.

## 1.4 Manfaat

Pada penelitian yang akan dilakukan sangat diharapkan akan memiliki manfaat yang berguna, manfaat tersebut telah dijabarkan pada poin-poin berikut ini:

1. Menyediakan teknologi atau sistem yang bisa mengklasifikasi informasi tempat di kota Malang khususnya tempat tinggal pada media sosial Twitter.
2. Mempermudah bagi perantau untuk mendapatkan informasi tentang tempat tinggal khususnya kost dan kontrakkan.

## 1.5 Batasan masalah

Pada penelitian yang akan dilakukan, permasalahan yang diangkat memiliki batasan-batasan tertentu yang telah dijabarkan sebagai berikut:

1. Penelitian yang akan dilakukan berfokus pada klasifikasi informasi tempat tinggal yang berada di kota Malang hanya pada Twitter.
2. *Dataset* berupa *tweet* dengan format data hanya teks yang berasal dari *tweet* pengguna Twitter yang sudah dipilih sesuai dengan kategori kontrakkan, kost putri, dan kost putra.
3. Jumlah *tweet* yang digunakan sebanyak 150 data latih dan 60 data uji.
4. Kategori kelas terdiri dari 3 kelas yaitu kontrakkan, kost putra, dan kost putri
5. Metode yang digunakan untuk klasifikasi adalah *Naïve Bayes.*
6. Fitur yang digunakan adalah *Information Gain* dan *Term Frequency-Inverse Document Frequency-Class Frequency*(TF-IDF-CF)
7. *Dataset* yang digunakan berasal dari *crawling* data dari Twitter pada pengguna tertentu
8. Penggunaan Python 3.6 sebagai bahasa untuk membangun kode program dengan memanfaatkan beberapa *library* yang dibutuhkan. Dalam implementasi sistem menggunakan pengembangan aplikasi Spyder Anaconda.

## 1.6 Sistematika pembahasan

Sistematika pembahasan untuk membuat dokumen dalam penelitian ini dapat dilihat berikut ini:

**BAB I – PENDAHULUAN**

Dalam pendahuluan berisi latar belakang yang menjadi dasar mengapa penelitian ini perlu dilakukan, rumusan masalah sebagai dasar diangkat

topik penelitian, tujuan yang ingin dicapai dari penelitian, manfaat yang ingin dicapai dari penelitian, batasan masalah sebagai aturan dalam mengerjakan penelitian dan sistematika penulisan dokumen

**BAB II – LANDASAN KEPUSTAKAAN**

Landasan kepustakaan berisi pembahasan mengenai penelitian-penelitian yang serupa dalam hal objek maupun metode-metode yang digunakan, bab ini juga berisi teori-teori yang bersangkutan dalam penelitian untuk memperkuat dan mendukung proses penelitian.

**BAB III – METODOLOGI**

Metodologi berisi tentang tahapan atau langkah-langkah dalam menyelesaikan penelitian yang akan dilakukan.

**BAB IV – PERANCANGAN**

Bab perancangan berisi tentang perancangan sistem yang akan dibuat, perancangan ini sendiri meliputi beberapa hal yaitu perancangan algoritme, perancangan perhitungan manual, dan perancangan pengujian

**BAB V – IMPLEMENTASI**

Bab ini adalah lanjutan dari bab perancangan dimana pada bab ini berisi tentang penerapan dari perancangan yang telah dibuat, atau bisa dibilang proses dalam membangun sistem pada penelitian.

**BAB VI – PENGUJIAN DAN ANALISIS**

Bab pengujian dan analisis berisi tentang hasil pengujian dari system yang telah dibuat, yang kemudian hasil tadi akan dianalisis untuk mengambil sebuah kesimpulan dari hasil pengujian.

**BAB VII – PENUTUP**

Bab penutup berisi tentang kesimpulan yang telah diambil atau bisa dibilang hasil akhir yang didapat tentang penelitian yang telah dilakukan serta berisi saran yang berguna untuk pengembangan pada penelitian selanjutnya

# BAB 2 LANDASAN KEPUSTAKAAN

## 2.1 Kajian Pustaka

Kajian pustaka berisi tentang penelitian-penelitian yang telah dilakukan sebelumnya yang memiliki hubungan dengan penelitian ini. Penelitian pada kajian pustaka ini berisi tentang teori atau metode yang memiliki hubungan dengan penelitian untuk permasalahan pada penelitian ini. Beberapa penelitian sebelumnya dapat dilihat pada pada Tabel dibawah.

**Tabel 2.1 Kajian Pustaka**

| No. | Judul dan Objek | Metode | Hasil |
|---|---|---|---|
| 1. | **Judul:**<br>*An improvement of TFIDF weighting in text categorization*(Liu & Yang, 2012)<br>**Objek:**<br>Pembobotan TD-IDF | *TD-IDF-CF* | Dihasilkan sebuah metode pembobotan TF-IDF yang dikembangkan dengan menambahkan sebuah parameter yaitu *class frequency,* yang mampu meningkatkan presisi untuk kategorisasi teks |
| 2. | **Judul:**<br>*Efficient Feature Subset Selection Algorithm for High Dimensional Data* (Chormunge, Jena & 2016)<br>**Objek:**<br>Seleksi Fitur | *Information Gain, Relief, Chi-Squared* | Algoritme seleksi fitur yaitu *Information gain* digunakan untuk memperoleh fitur yang paling optimal sehingga mampu mengefisienkan waktu dan meningkatkan performa komputasi |
| 3. | **Judul:**<br>Penerapan Metode Content-Based Filtering Pada Sistem Rekomendasi Kegiatan Ekstrakulikuler (Studi Kasus di Sekolah ABC) (FirmahSyah & Gantini, 2016) | *Information Gain, Naïve Bayes Classifier* | Hasil penelitian ini menunjukkan bahwa algoritme NBC dengan seleksi fitur *information gain* memiliki akurasi yang lebih baik dibandingkan dengan algoritme NBC tanpa |

| | | | |
|---|---|---|---|
| | **Objek:**<br>Kegiatan Ekstrakulikuler | | menggunakan seleksi fitur |
| 4. | **Judul:**<br>Klasifikasi Teks Dengan Naïve Bayes Classifier (NBC) Untuk Pengelompokan Teks Berita Dan Abstract Akademis (Hamzah, 2012)<br>**Objek:**<br>Teks Berita dan Akademis | *Naïve Bayes Classifier* | Hasil akhir penelitian memiliki tingkat akurasi yang tinggi dimana *precision* sekitar 98,42%, sedangkan *recall*-nya 93,5%. |
| 5. | **Judul:**<br>*Lyrics Classification using Naive Bayes* (Bužić & Dobša, 2018)<br>**Objek:**<br>Lirik Lagu | *Naïve Bayes Classifier* | Penelitian ini memperlihatkan Naïve Bayes mampu menjadi sebuah classifier yang sangat baik dengan hasil evaluasi precision senilai 0,93, *recall* senilai 0,95 dan *F-Measure* senilai 0,94 |
| 6. | **Judul:**<br>*Implementation Of Naive Bayes Classifier Algorithm On Social Media (Twitter) To The Teaching Of Indonesian Hate Speech* (Fatahillah, Suryati dan Haryawan, 2018)<br>**Objek:**<br>*Indonesian Hate Speech* | *Naïve Bayes Classifier* | Sistem yang dibuat mampu menghasilkan akurasi 93 % untuk mengklasifikasikan *Hate Speech* berbahasa Indonesia |

## 2.2 Dasar Teori

Dasar teori berisi tentang penjelasan tentang teori-teori dan metode yang berhubungan dengan penelitian

### 2.2.1 Kos dan Kontrakan

Menurut KBBI kos adalah bentuk tidak baku dari indekos yang memiliki makna "tinggal di rumah orang lain dengan atau tanpa makan (dengan membayar setiap bulan)", jadi bisa dibilang kos merupakan tempat tinggal sementara bagi seseorang yang belum memiliki tempat tinggal di suatu tempat. Tak jauh beda dengan kontrakan bisa dibilang kontrakan dengan kos memiliki beberapa perbedaan dimana biasanya kontrakan memiliki jangka waktu 1-2 tahun dengan perjanjian kontrak antara pemilik dan penyewa kontrakan, dan biasanya disewakan seisi rumah berbeda dengan kos yang biasanya hanya 1 kamar

### 2.2.2 Twitter

Twitter adalah sebuah situs jejaring sosial yang sedang berkembang pesat saat ini karena pengguna dapat berinteraksi dengan pengguna lainnya dari komputer ataupun perangkat mobile mereka dari manapun dan kapanpun. Terhitung pada Januari 2011, terdapat hamper 200 juta pengguna Twitter, yang setiap harinya mempos sebanyak 110 juta *tweet*.(Chiang, 2011).

Twitter sendiri dapat digunakan untuk berinteraksi dengan keluarga, rekan kerja, hingga teman. Twitter memberikan akses kepada pengguna untuk mengirimkan sebuah pesan singkat atau biasa disebut *tweet* yang memiliki batasan karakter maksimal yaitu sebanyak 280 karakter. *Tweet* ini yang menjadi wadah interaksi antar pengguna Twitter dengan cara mengirimkan berita, apa yang sedang dilakukan, atau tentang kejadian yang baru saja terjadi.

### 2.2.3 *Text Mining*

Text mining merupakan proses untuk mendapatkan sebuah informasi dan pengetahuan yang berguna yang biasa disebut dengan proses ekstraksi pola, dimana proses ini dilakukan pada data yang tidak terstruktur. *Text mining* dan data mining memiliki kesamaan pada proses dan tujuan, namun hanya berbeda pada jenis masukan yang berbeda, pada text mining memiliki masukan berupa data yang tidak terstruktur, seperti PDF, teks, Word, dll (Feldman dan Sanger, 2007). Text mining memiliki dua tahap proses yaitu, penerapan struktur pada sumber teks yang kemudian akan dilanjutkan pada proses ekstraksi pola itu sendiri. Beberapa contoh penerapan Text mining yang popular adalah:

1. Ekstraksi Informasi (*Information Extraction*) adalah cara atau proses untuk mengidentifikasi frasa dalam kata kunci dan juga hubungannya dalam teks setelah melakukan pencocokan pola untuk melihat urutan tertentu.
2. Pelacakan Topik (*Topic Tracking*) adalah pencarian dokumen yang tepat dengan seseorang pengguna dimana pencariannya didasarkan pada profil dan dokumen, dilihat dari pengguna yang bersangkutan.
3. Perangkuman (*Summarization*) adalah membuat ringkasan dengan hanya mengambil informasi penting dalam sebuah dokumen, biasanya peringkasan ini maksimalnya hanya setengah dari dokumen asli.

4. Pengelompokkan (*Clustering*) adalah proses pengelompokkan dokumen berdasarkan kemiripan yang ada pada dokumen tersebut, tanpa adanya pelabelan kategori sebelumnya.
5. Klasifikasi (*Classification*) adalah mengelompokkan dokumen kedalam sebuah kategori dengan mencocokkan dengan dokumen yang sudah terdapat pada kategori yang sudah ditentukan sebelumnya.
6. Penjawaban Pertanyaan (*Question Answering*) adalah memberikan jawaban secara otomatis dari pertanyaan yang diberikan, pencarian jawaban berdasar pada dokumen latih yang sudah ada.

### 2.2.4 *Text Preprocessing*

*Text preprocessing* merupakan tahap persiapan data berupa teks sebelum diproses pada tahap selanjutnya, proses ini membuat data menjadi data yang lebih terstruktur dan siap diolah, ada beberapa tahap pada proses preprocessing sesuai dengan kebutuhan sistem. Namun pada umumnya preprocessing memiliki beberapa tahap seperti berikut:

1. *Case Folding*

   *Case folding* merupakan tahap untuk mengubah seluruh huruf pada dokumen menjadi huruf kecil.(pembatas)(Triawati, 2009).

2. *Tokenizing*

   Tahap *tokenizing / parsing* adalah proses memecah teks pada dokumen menjadi sebuah kata (Triawati, 2009). Selain itu, spasi merupakan pemisah antar kata tersebut.

3. *Filtering*

   Tahap filtering merupakan tahap yang berguna untuk menyeleksi kata-kata penting dalam dokumen. Proses ini biasanya menggunakan data berupa stopword yang berisi kata-kata yang tidak signifikan atau tidak penting pada dokumen. (Triawati, 2009).

4. *Stemming*

   *Stemming* merupakan tahap untuk mengubah kata yang ada pada seluruh dokumen menjadi kata dasar dengan menggunakan aturan tertentu. (Triawati, 2009).

### 2.2.5 *Term Frequency-Inverse Document Frequency(TF-IDF)*

TF-IDF adalah sebuah metode yang biasa digunakan untuk pembobotan,pembobotan ini pertama kali diperkenalkan pada sistem temu kembali informasi atau biasa disebut *information retrieval*. TF-IDF adalah pembobotan dimana bobot sebuah kata berdasarkan frekuensi dokumen terbalik.

Yang berarti jika sebuah kata semakin banyak muncul pada banyak dokumen, maka kata tersebut memiliki bobot yang lebih kecil (Liu dan Yang, 2012). Pembobotan TF-IDF dapat dilihat pada Persamaan 2.1:

$$w_{ij} = tf_{ij} * lo\ g\left(\frac{N}{n_j}\right) \quad (2.1)$$

Keterangan:

$w_{ij}$ = Bobot kata *j* pada dokumen *i*

$tf_{ij}$ = Frekuensi kata *j* pada dokumen *i*

$N$ = Jumlah dokumen

$n_j$ = Jumlah dokumen yang terdapat kata *i*

Namun persamaan di atas terkadang memiliki masalah yaitu ketika nilai N sama dengan $n_j$ bobot akan bernilai 0, sehingga diubah menjadi seperti Persamaan 2.2:

$$w_{ij} = \log(tf_{ij} + 1) * lo\ g\left(\frac{N+1}{n_j}\right) \quad (2.2)$$

#### 2.2.5.1 *Term Frequency-Inverse Document Frequency-Class Frequency* (TF-IDF-CF)

TF-IDF-CF adalah sebuah metode pengembangan dari TF-IDF, dimana ditambahkan sebuah parameter untuk merepresentasikan karakteristik tiap kelasnya, parameter ini disebut *class frequency*, yang berfungsi untuk menghitung frekuensi kata pada dokumen pada suatu kelas. Rumus TF-IDF-CF dapat dilihat pada persamaan 2.3:

$$w_{ij} = \log(tf_{ij} + 1) * lo\ g\left(\frac{N+1}{n_j}\right) * \frac{n_{cij}}{N_{ci}} \quad (2.3)$$

Keterangan:

$N_{ci}$ = Jumlah dokumen pada kelas *c* untuk dokumen *i* pada kelasnya

$n_{cij}$ = Jumlah dokumen dimana kata *j* muncul pada kelas c pada dokumen *I* pada kelasnya

### 2.2.6 *Information Gain*

*Information gain* (IG) adalah metode seleksi fitur sederhana dengan cara memeringkat atribut, metode ini biasanya digunakan pada aplikasi kategorisasi teks, analisis data *microarray*, dan analisis data citra (Chormunge dan Jena, 2016). Rumus untuk menghitung *entropy* dapat dilihat pada persamaan 2.4, setelah didapatkan nilai entropy maka dapat dihitung nilai IG pada persamaan 2.5 (Firmansyah dan Gantini, 2016).

$$Entropy(S) = \sum_i^c -P_i \, log_2 P_i \quad (2.4)$$

Dimana *c* adalah jumlah nilai yang ada pada kelas klasifikasi dan $P_i$ merupakan jumlah sampel untuk kelas *i*

$$Gain(S, A) = Entropy(S) - \sum_{Values(a)} \frac{|S_v|}{|S|} Entropy(S_v) \quad (2.5)$$

Dengan A merupakan atribut, v adalah nilai yang mungkin untuk atribut A, Values(A) adalah himpunan nilai-nilai yang mungkin untuk A, $|S_v|$ adalah jumlah sampel untuk nilai v, |S| merupakan jumlah sampel data dan *Entropy($S_v$)* adalah *entropy* untuk sampel yang dimiliki nilai v. Namun untuk seleksi fitur pada pengklasifikasian teks terdapat perbedaan rumus untuk perhitungan IG dari teks dituliskan pada Persamaan 2.6 (Gede Widnyana Putra, Sudarma, dan Satya Kumara, 2016).

$$IG(t) = -(\sum_{i=1}^{m} P\gamma(ci) \log P\gamma(ci) + P(t) \sum_{i=1}^{m} P\gamma(ci|t) \log P\gamma(ci|t) + P\gamma(\bar{t}) \sum_{i=1}^{m} P\gamma(ci|\bar{t}) \log P\gamma(ci|\bar{t}) \quad (2.6)$$

Dimana $P_\gamma(c_i)$ adalah pelung sebuah dokumen pada kelas tertentu, $P_\gamma(t)$ adalah peluang term t yang muncul pada dokumen, $P_\gamma(\bar{t})$ adalah peluang term *t* yang tidak muncul pada dokumen, $P_\gamma(c_i|t)$ adalah peluang kemunculan term *t* didalam dokumen pada kelas tertentu, sedangkan $P_\gamma(c_i|\bar{t})$ adalah peluang ketidakmunculan term t didalam dokumen pada kelas tertentu.

### 2.2.7 *Naive Bayes Classifier*(NBC)

Metode NBC memiliki dua tahap, yaitu tahap pelatihan dan tahap klasifikasi. Pada tahap pertama yaitu tahap pelatihan, tahap ini dilakukan proses analisis terhadap sebuah dataset yang berupa dokumen, analisis ini dilakukan untuk mendapatkan pola dan kata-kata yang mungkin merepresentasikan suatu dokumen. Pada tahap ini juga dilakukan penentuan probabilitas prior untuk setiap kategori berddasarkan sampel dokumen. Kemudian pada tahap klasifikasi ditentukan nilai kategori dari suatu dokumen berdasarkan kata yabg muncul dalam dokumen klasifikasi (Hamzah, 2012). Secara garis besar rumus untuk *Naïve Bayes Classifier* sebagai berikut.

$$P(c_j|w_i) = \frac{P(c_j)\, P(w_i|c_j)}{P(w_i)} \quad (2.7)$$

Keterangan:

$P(c_j|w_i)$ : Peluang kategori *j* ketika terdapat kemunculan kata *i*

$P(w_i|c_j)$ : Peluang sebuah kata *i* masuk ke dalam kategori *j*

$P(c_j)$ : Peluang kemunculan sebuah kategori *j*

$P(w_i)$ : Peluang kemunculan sebuah kata

Pada proses klasifikasi peluang kemunculan kata dapat dihilangkan, karena pada perbandingan hasil kategori dari setiap kategori peluang tersebut tidak berpengaruh. Sehingga pada proses klasifikasi dapat disederhanakan dengan Persamaan (2.8).

$$P(c_j|w_i) = P(c_j)\, P(w_i|c_j) \tag{2.8}$$

Perhitungan prior digunakan untuk menghitung peluang kemunculan kategori pada semua dokumen, perhitungan prior dapat dilihat pada Persamaan (2.9).

$$P(c_j) = \frac{N_c}{N} \tag{2.9}$$

Keterangan:

$N_c$ : Banyak dokumen berkategori $c_j$ pada dokumen latih

*N* : Jumlah keseluruhan dokumen latih yang digunakan

Perhitungan *posterior* merupakan perhitungan yang dilakukan dengan mengalikan *prior* dengan *total conditional probability*. Rumus perhitungan dapat dilihat pada Persamaan (2.10).

$$P(c_j|w_i) = P(c_j)\ x\ P(w_1|c_j)\ x\ P(w_2|c_j) \ldots P(w_n|c_j) \tag{2.10}$$

#### 2.2.7.1 *Gaussian Naïve Bayes*

Pada penelitian ini pada pembobotan kata menggunakan TF-IDF-CF sehingga memiliki nilai yang *continuous,* sehingga metode *Naïve Bayes* yang digunakan adalah *Gaussian Naïve Bayes,* metode ini merupakan salah satu model dari metode *naïve bayes* dengan melakukan perhitungan data kontinu, dimana setiap tipe nya mencirikan *Multivariate Gaussian* dan Normal *Probability Density Function*(PDF) (Capdevila dan Flórez 2009). Metode ini memiliki 2 parameter yang perlu didapatkan terlebih dahulu sebelum mampu menghitung nilai *likelihood*-nya, yaitu nilai rata-rata dan varians. Persamaan dari *Gaussian Naïve Bayes* dapat dilihat pada Persamaan 2.11.

$$P(w_i|c) = x = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp^{\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right)} \tag{2.11}$$

$P(w_i|c)$ : Peluang sebuah kata *i* masuk ke dalam setiap kategori

$\mu_y$ : Mean dari data latih

$\sigma_y^2$ : varian dari data latih

### 2.2.8 Akurasi

Pengujian akurasi adalah langkah yang dilakukan untuk melihat persentase ketepatan yang dihasilkan sistem dibandingkan dengan data uji yang sebenarnya. Akurasi dapat dihitung menggunakan persamaan pada Persamaan 2.8.

Secara umum proses dari pengujian akurasi dapat dilihat pada Persamaan:

$$Akurasi = \frac{jumlah\ klasifikasi\ benar}{jumlah\ data\ uji} \times 100\% \quad (2.12)$$

Pada tahap pengujian ini dilakukan 5 pengujian akurasi yang berbeda yaitu dengan mengombinasikan proses pembobotan TF-IDF-CF dan seleksi fitur IG, kemudian akan melakukan pengujian tanpa menggunakan fitur angka yang akan dilakukan setelah mendapatkan threshold information gain optimal paling optimal, dimana 5 pengujian itu yaitu dengan TF-IDF-CF dan IF, dengan TF-IDF-CF dan tanpa IG, tanpa TF-IDF-CF dan dengan IG, tanpa TF-IDF-CF dan tanpa IG, dan pengujian dengan IG optimal dan metode pembobotan terbaik tanpa menggunakan fitur angka.

# BAB 3 METODOLOGI

## 3.1 Tipe Penelitian

Tipe penelitian ini adalah penelitian non implementatif dengan jenis analitik, yang artinya penelitian ini berfokus pada investigasi atau pengamatan terhadap sesuatu situasi tertentu yang akhirnya menghasilkan tinjauan ilmiah. Sedangkan analitik artinya penelitian ini berusaha untuk menjelaskan hubungan antar elemen dalam objek penelitian dengan domain masalah yang diteliti, pada kasus ini domain masalah penelitian yang diangkat adalah dari tempat tinggal di kota Malang.

## 3.2 Strategi Penelitian

Strategi penelitian pada penelitian seleksi fitur IG untuk klasifikasi tempat tinggal di Kota Malang berdasarkan *tweet* menggunakan metode *Naïve Bayes* dan pembobotan TF-IDF-CF adalah menggunakan studi kasus, yang artinya disini penelitian bertujuan untuk memahami dan menjelaskan objek yang diteliti.

## 3.3 Peralatan Pendukung

Peralatan pendukung adalah perangkat keras atau lunak yang digunakan dalam penelitian ini. Peralatan yang pertama adalah perangkat keras dapat dilihat sebagai berikut:

- Intel® Core™ i5-6200U CPU @ 2.30GHz (4 CPUs), ~2.40Ghz
- RAM 4 GB
- Harddisk 500 GB

Peralatan perangkat lunak dapat dilihat sebagai berikut:

- Sistem Operasi Windows 10 Pro 64-bit
- Editor Spyder Anaconda Python 2.7
- Database MongoDB

## 3.4 Teknik Pengumpulan Data

Untuk penelitian ini data yang digunakan adalah data primer, yaitu data diambil secara langsung melalui akun Twitter tertentu, pada kasus ini akun Twitter yang digunakan sebagai sumber adalah akun @infokostmalang. Diambil menggunakan crawler dengan API Twitter. Setelah data sudah di crawl data akan di filter secara manual oleh peneliti untuk memilih data yang valid dan sesuai dengan penelitian, yaitu data yang mencakup kategori kost putra, kost putri, dan kontrakan.

## 3.5 Data Penelitian

Pada penelitian ini data yang digunakan adalah data *tweet* berbahasa Indonesia yang didapatkan melalui crawling data, data hasil crawling yang

didapatkan sejumlah 2148 data, setelah dilakukan *filtering* secara manual didapatkan data valid sebanyak 259 data yang terbagi menjadi 3 kategori yaitu, 50 untuk kategori kontrakan, 63 untuk kategori kost putra, dan 146 untuk kategori kost putri. Untuk menjadikan data seimbang untuk pelatihan digunakan 150 data yaitu 50 data untuk setiap kategori, dari data yang ada tadi dilakukan pelatihan dengan menggunakan *Naïve Bayes,* kemudian digunakan sejumlah 60 data uji untuk melakukan pengujian sistem yang telah dibuat.

## 3.6 Teknik Analisis Data

Pada kasus ini Teknik analisis data ini dengan menggunakan pengujian berupa menghitung hasil akurasi sistem yang telah dibuat, terdapat 5 hasil perhitungan akurasi dengan mengombinasikan penggunaan pembobotan TF-IDF-CF dan seleksi fitur IG, yang nantinya masing-masing hasil akurasi akan dibandingkan untuk menentukan yang mana yang terbaik.

## 3.7 Perancangan Algoritme

Pada penelitian rancangan algoritme yang digunakan ada beberapa tahap, tahapn-tahapnnya yaitu ada, proses *preprocessing* data, proses ini bermaksud untuk membersihkan data dan menyiapkan data sebelum di proses, kemudian dilakukan seleksi fitur dengan menggunakan metode IG yang berfungsi untuk memilih fitur-fitur mana saja yang akan digunakan dalam pengklasifikasian, kemudian dilakukan pembobotan kata menggunakan TF-IDF-CF untuk mendapatkan bobot nilai setiap fiturnya, setelah itu dilakukan proses klasifikasi menggunakan metode Naïve Bayes untuk mendapatkan kelas atau kategori dari masing-masing *tweet.*

# BAB 4 PERANCANGAN

## 4.1 Deskripsi Sistem

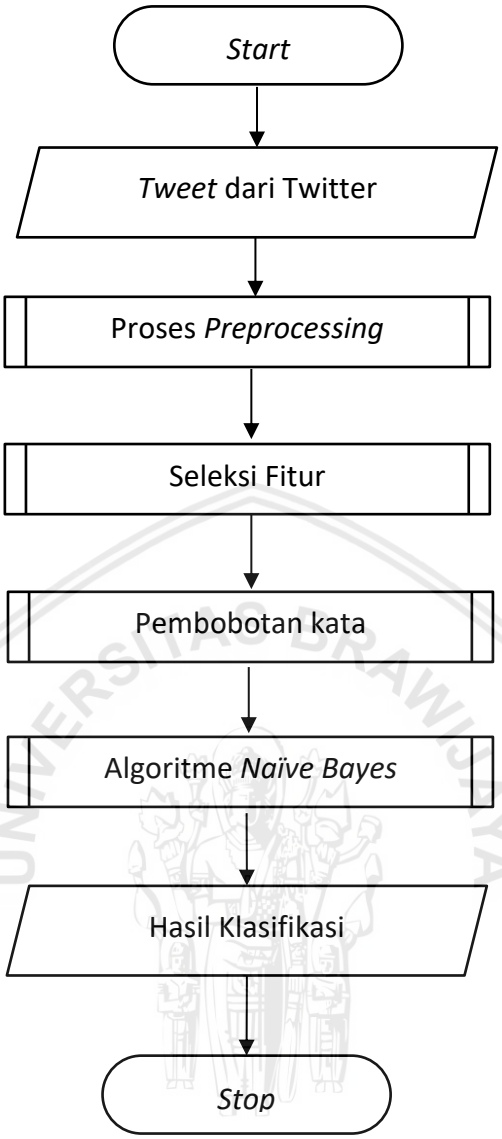
Sistem yang akan dibuat pada penelitian ini dibangun menggunakan Bahasa pemrograman Python pada implementasinya. Tujuan utama sistem ini dibuat adalah untuk mengklasifikasikan informasi *tweet* pada Twitter apakah termasuk pada kategori kost putra, kost putri, atau kontrakkan. Untuk kontribusi, penelitian ini akan menguji teori penggunaan pembobotan TF-IDF-CF dan seleksi fitur IG pada klasifikasi teks menggunakan metode Naïve Bayes. Manfaat utama dari penelitian ini adalah sarana untuk membantu mahasiswa baru dalam memenuhi kebutuhan utamanya dan mencari informasi spesifik mengenai tempat tinggal sementara di kota Malang.

## 4.2 Persiapan Data

Dalam penelitian ini data yang digunakan merupakan data primer di mana data didapatkan dengan melakukan *crawling* menggunakan *API* Twitter pada akun @infomalang dan @infokostmalang. Dari data yang di *crawl* tadi memiliki beberapa atribut, seperti id_str, name, screen_name, user_id_str, text, created_at, replies_tweet, oa, kelas, isItOK, isInfo. Setelah data *tweet* tadi didapatkan maka akan dilakukan *filtering* dan pelabelan manual oleh peneliti, dimana *filtering* ini dilakukan untuk memilih tweet yang sekiranya bisa dikatakan benar dan valid untuk diproses, kemudian data akan dilabeli manual terdapat 3 label yaitu kontrakan, kost putra, dan kost putri. Total data yang didapatkan dari *crawling* sejumlah 2148, setelah dilakukan *filtering* dan pelabelan data yang bisa dikatakan valid sejumlah 259 dengan jumlah perkategori yaitu, 50 untuk kategori kontrakan, 63 untuk kategori kost putra, dan 146 untuk kategori kost putri. Untuk keseimbangan data untuk data latih maka data yang digunakan untuk data latih sejumlah 50 untuk tiap kategorinya, sehingga jumlah data yang digunakan untuk data latih sejumlah 150. Data latih yang sudah terpilih tadi akan di proses melalui pembelajaran *Naïve Bayes* dengan menghitung frekuensi kemunculan sebuah kata pada dokumen di kelas tertentu. Sedangkan untuk pengujian digunakan 60 data dengan pembagian 45 diuji dari *tweet* asli dan 15 dari *tweet* baru.

## 4.3 Perancangan Proses

Pada perancangan proses ini akan menjelaskan proses klasifikasi yang dibagi pada beberapa tahap, pada tahap pertama sistem akan menerima masukkan berupa sebuah *tweet* berbahasa Indonesia dari Twitter yang akan dilakukan *preprocessing* untuk menyiapkan data sebelum diolah, kemudian akan dilakukan seleksi fitur atau term yang terdapat pada tweet tersebut dengan menggunakan metode IG, kemudian term atau kata akan dilakukan pembobotan, yang kemudian hasil pembobotan akan masuk kedalam algoritme *Naïve* Bayes untuk dilaukkan klasifikasi, setelah itu akan didapatkan ahsil klasifikasi dari *tweet* yang telah dimasukkan 4.2.

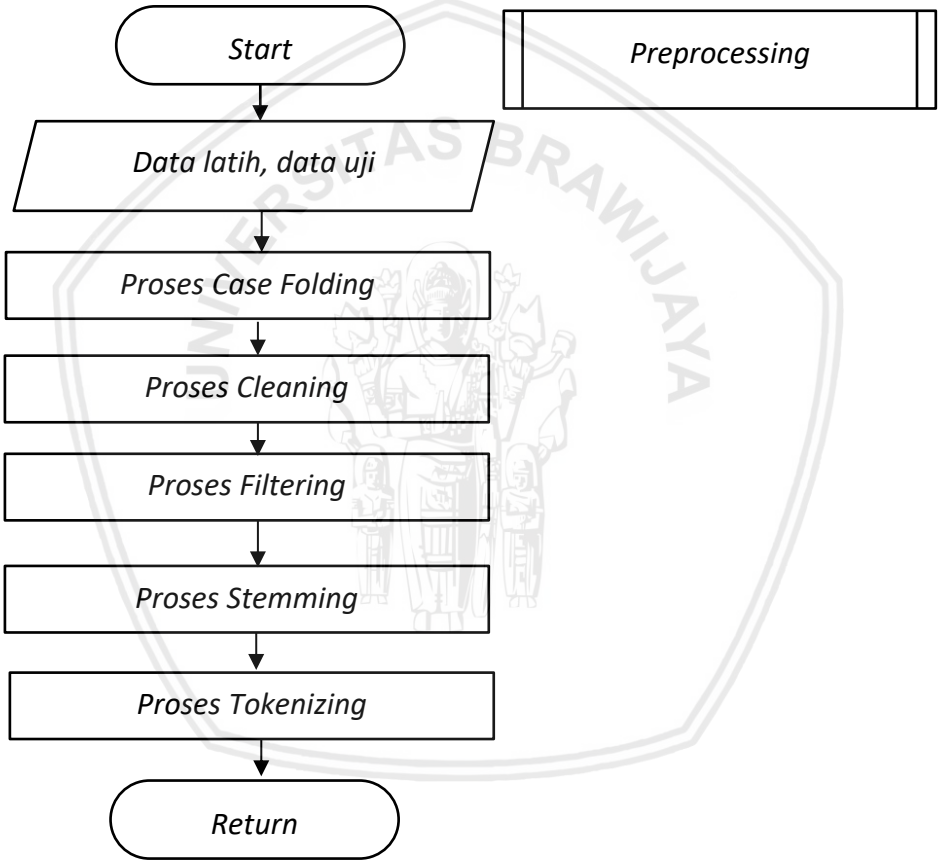



**Gambar 4.1 Diagram Alir Sistem**

### 4.3.1 Proses Preprocessing

*Preprocessing* dilakukan untuk mempersiapkan data sebelum diproses lebih lanjut.Pada sistem yang ini ada beberapa tahapan dalam *preprocessing* yang dilakukan yang pertama adalah melakukan *case folding* yang berfungsi untuk mengubah seluruh huruf kapital menjadi huruf kecil, setelah itu akan dilakukan cleaning proses ini berfungsi untuk menghilangkan membersihkan *tweet* dari karakter atau pola karakter yang tidak diperlukan pola karakter ini sendiri seperti URL, *hashtag*, dan nama pengguna yang biasa menggunakan @. Setelah itu dilakukan proses *filtering* atau yang biasa disebut *stopword removal* proses ini berguna untuk menghilangkan kata yang tidak penting atau bisa disebut *stopword*, contohnya seperti yang, dan, di dll. Kemudian ada proses *stemming* yang

merupakan proses untuk mengubah sebuah kata menjadi kata dasar. Yang terakhir ada proses *Tokenizing*, pada sistem ini tokenizing atau tokenisasi memang dilakukan pada bagian akhir *preprocessing*, *tokenizing* ini adalah proses memecah sebuah kalimat menjadi term proses ini dilakukan terakhir karena pada penelitian ini peneliti menggunakan *API* pujangga *InaNLP* pada beberapa tahapan proses *preprocessing* yaitu pada proses *filtering* dan *stemming* hal ini mengakibatkan untuk menggunakan API ini sistem perlu memanggil *API* untuk melakukan proses tersebut sebanyak dokumen yang akan di proses, jika tokenisasi dilakukan di awal maka sistem akan memanggil *API* sebanyak term yang sudah ditokenisasi yang tentu akan memakan waktu yang jauh lebih banyak dibandingkan dengan memanggil sebanyak jumlah dokumen yang akan di proses. Berikut ini merupakan diagram alir dari proses *preprocessing* pada Gambar 4.3.

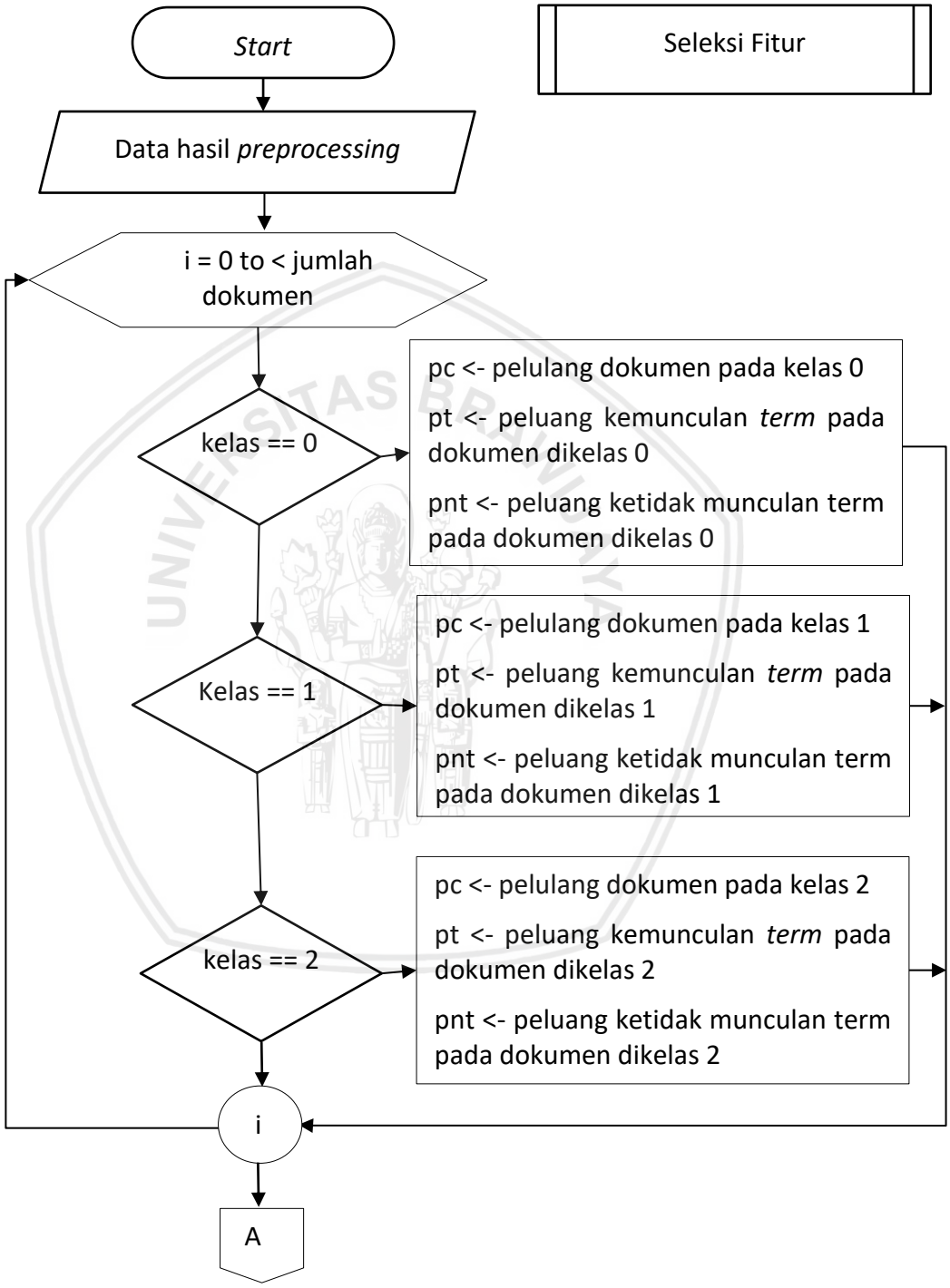


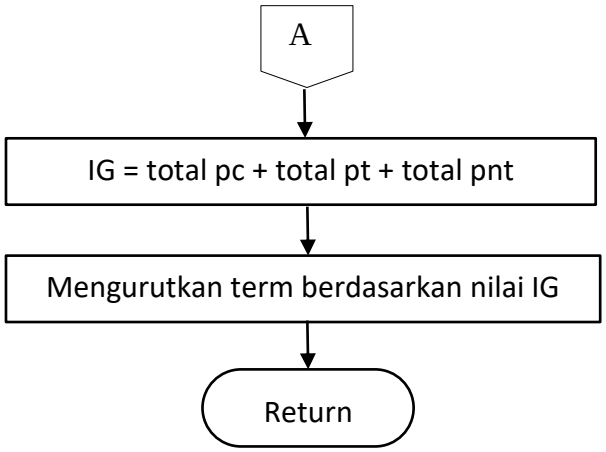

**Gambar 4.2 Diagram Alir Proses *Preprocessing***

### 4.3.2 Seleksi Fitur

Pada seleksi fitur berfungsi untuk memilih fitur yang akan digunakan pada proses selanjutnya pada penilitian ini metode yang digunakan adalah IG, cara kerja metode ini adalah yang pertama menghitung peluang dokumen pada setiap kelas, kemudian menghitung kemunculan sebuah *term* pada dokumen di kelas tertentu, yang terakhir akan menghitung ketidak munculan sebuah *term* pada dokumen di kelas tertentu, setelah dihitung maka nilai dari 3 proses perhitungan tadi akan

dijumlahkan untuk mendapatkan nilai IG dari setiap *term* nya, kemudian akan di urutkan berdasarkan nilai IG tertinggi dan akan dipilih sejumlah *term* dengan nilai tertinggi untuk digunakan dalam proses selanjutnya

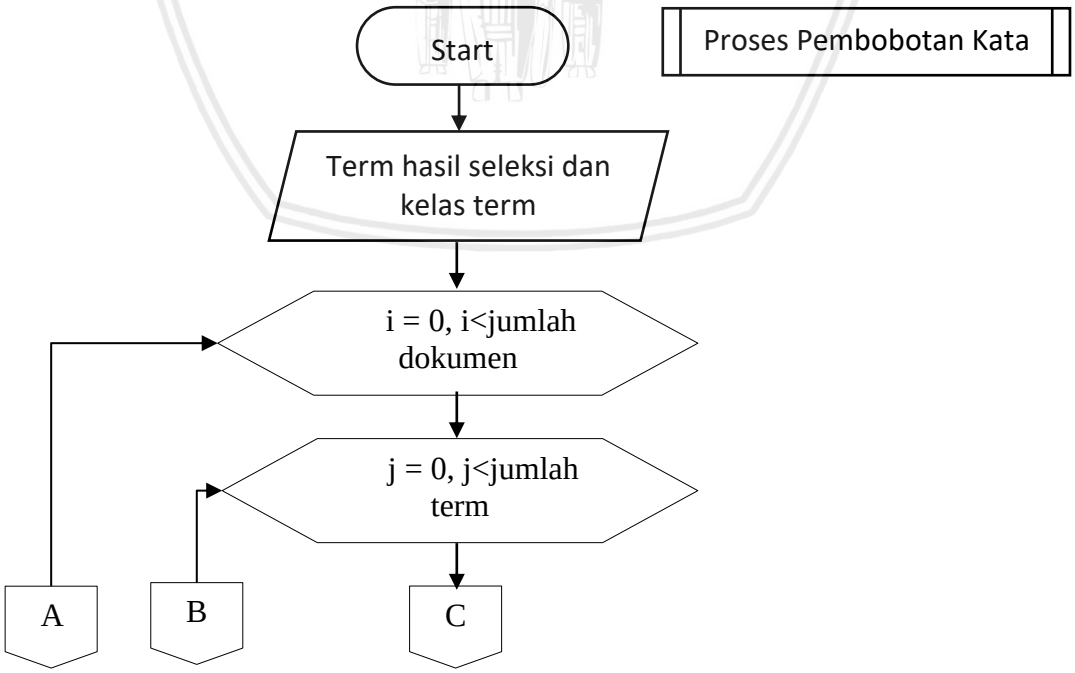


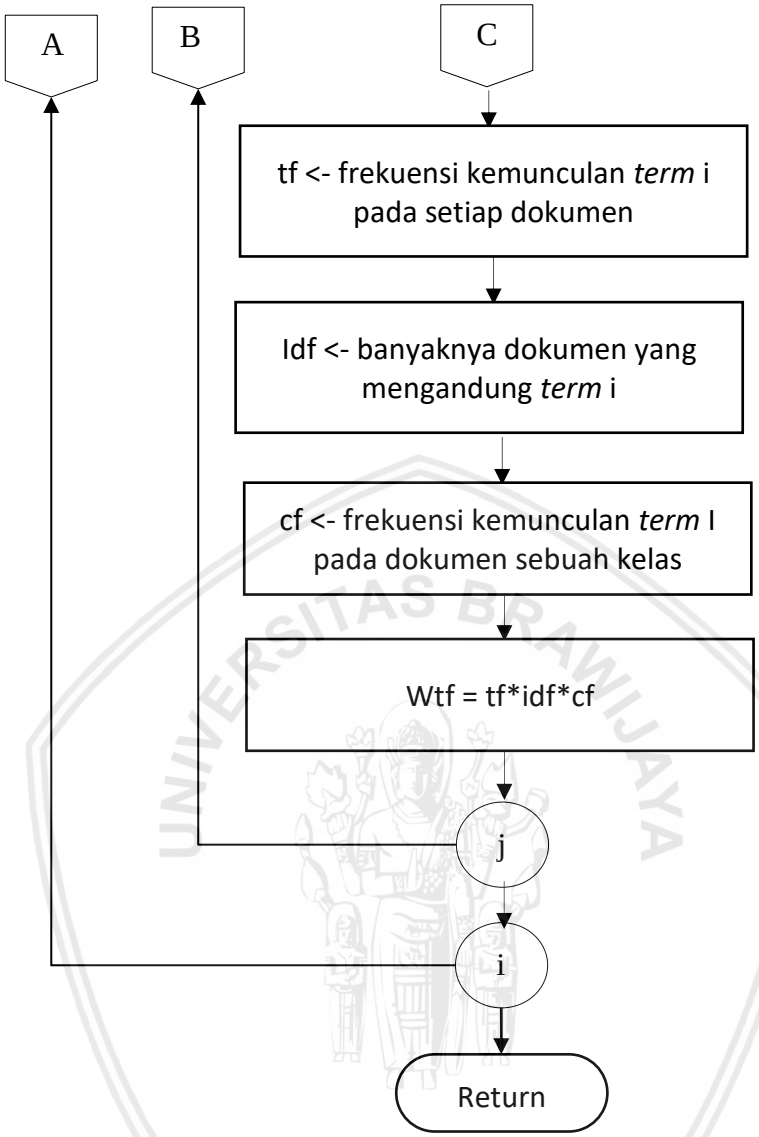



**Gambar 4.3 Diagram Alir Seleksi Fitur**

### 4.3.3 Proses Pembobotan Kata

Proses pembobotan kata merupakan proses untuk memberi bobot pada setiap kata atau term yang ada pada data latih, pembobotan ini sendiri nanti akan dipengaruhi dari hasil seleksi fitur karena hanya akan ada beberapa term yang digunakan dan dihilangkan akibat metode seleksi fitur, pada penelitian ini pembobotan kata yang digunakan adalah TF-IDF-CF, TF-IDF CF adalah sebuah metode yang dikembangkan dari metoe TF-IDF dengan menambahkan parameter baru yaitu *class frequency*(CF) yang akan mengakibatkan nilai term akan semakin tinggi ketika semakin sering muncul pada semakin sedikit kelas. Berikut ini merupakan diagram alir dari proses pembobotan kata pada Gambar 4.5.



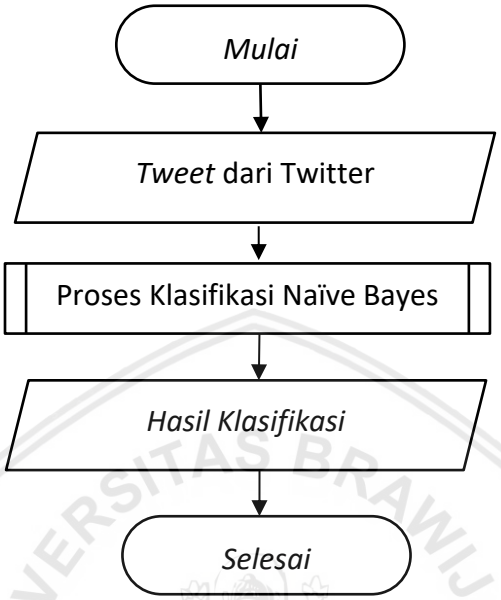



**Gambar 4.4 Diagram Alir Proses Pembobotan Kata**

### 4.3.4 Algoritme Naïve Bayes

Algoritme *Naive Bayes* merupakan tahapan yang dapat digunakan pada pemrosesan teks, khususnya dalam hal klasifikasi dokumen. Dalam proses ini terdapat beberapa tahapan, diantaranya adalah menghitung nilai probabilitas *prior*, nilai probabilitas *prior* didapatkan dari perhitungan probabilitas pembobotan kata dari kelas kontrakan, kelas kost putra dan kelas kost putri. Kemudian tahapan selanjutnya adalah *conditional probability*, tahap ini dilakukan dengan cara menghitung frekuensi kemunculan kata pada tiap dokumen yang kemudian akan dipergunakan untuk mencari nilai *Posterior*. Tahap *Posterior* diakukan dengan mengalikan hasil perhitungan *conditional probability* dari masing-masing kata pada suatu kelas. Dan hasil dari perkalian tersebut dilakukan

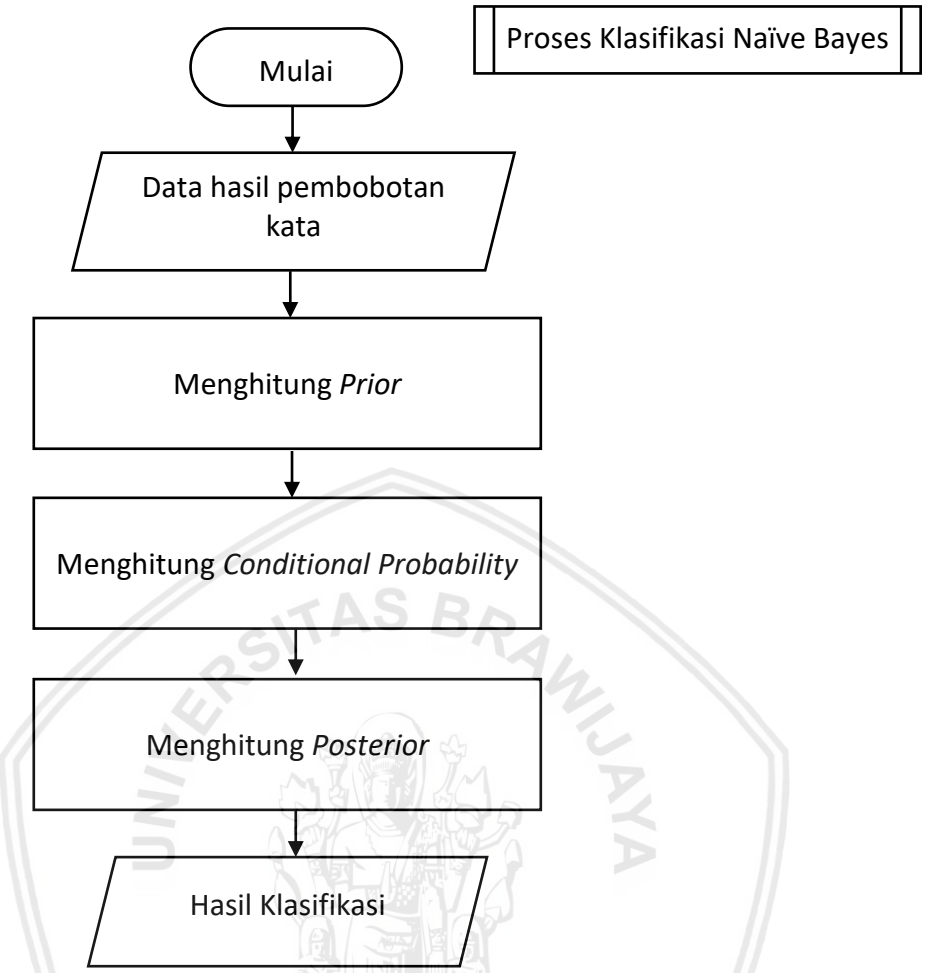
perbandingan untuk mencari nilai tertinggi. Nilai tertinggi tersebut akan menentukan sebuah dokumen masuk ke dalam kelas kontrakan, kost putra ataupun kost putri. Penjelasan alir proses dari penyelesaian metode klasifikasi *Naive Bayes* ditunjukkan pada Gambar 4.6.




**Gambar 4.5 Diagram Alir Algoritme Naïve Bayes**

### 4.3.5 Proses Klasifikasi Naïve Bayes

Proses Klasifikasi Naïve Bayes berguna untuk mendapat hasil klasifikasi dari data uji, proses ini pertama-tama menerima *input* berupa data uji yang sudah ditokenisasi dan data yang sudah di *training* sebelumnya. Data uji tadi kemudian akan disimpan kedalam sebuah *dictionary* yang kemudian akan masuk kedalam proses perhitungan *Naïve Bayes.* Perhitungan Naïve Bayes ini sendiri menghitung peluang sebuah dokumen masuk ke dalam sebuah kelas, dengan cara menghitung nilai kemunculan frekuensi setiap kata dari data uji tiap kelasnya dibagi dengan jumlah kata pada tiap kelas data latih ditambah dengan jumlah seluruh kata pada data latih, yang data uji akan masuk ke dalam kelas dengan nilai peluang tertinggi. Berikut ini merupakan diagram alir dari proses pengujian pada Gambar 4.7.




**Gambar 4.6 Diagram Alir Proses Klasifikasi Naïve Bayes**

## 4.4 Perhitungan Manual

Sub bab ini akan menjelaskan dan memperlihatkan perhitungan manual yang diharapkan bisa memberikan gambaran umum terkait alur proses perhitungan yang akan dibuat pada sistem.

Untuk langkah-langkah perhitungan manual sistem adalah sebagai berikut:

**Langkah 1.** Menentukan dataset yang dipakai

Pada proses manualisasi dataset yang digunakan hanya menggunakan 5 sampel dari keseluruhan data pada dataset meliputi data latih dan data uji dengan ketentuan:

a) Dataset merupakan kumpulan *tweet* menggunakan bahasa Indonesia
b) Terdapat 3 atribut dalam dataset meliputi label atau kelas dan *tweet*
c) Terdapat 3 kelas data yakni kontrakan dengan nilai 0, kost putra dengan nilai 1 dan kost putri dengan nilai 2.

d) Terdapat 6 data latih dan 1 data uji

Dataset yang akan digunakan dalam proses manualisasi ini dapat dilihat pada Tabel 4.1.

**Tabel 4.1 *Dataset Tweet* Tempat Tinggal**

| No | Label | Tweet |
|---|---|---|
| 1 | 0 | DIKONTRAKKAN<br>Rumah baru dekat kampus Universitas Brawijaya,Polinema,Unisma,UM,Jalan soehatt<br>(Jl. Dewandaru Dalam 37 C)<br><br>Monggo hub. ☎ 085649691086 (WA)<br>#infokostmalang #infomalang<br>cc @infokostmalang @infomalangraya @pojokpolinema @infomalang<br><br>#infokostmalang #dekatkampus https://t.co/8mkxWdTo53 |
| 2 | 0 | DIKONTRAKKAN rumah di daerah tidar, 6 kamar tidur, 1 kamar mandi, jemuran di atas. Dekat ke STIKI, ub, um, uin, ITN..<br>Monggo yg minat bisa hub 081259809311<br><br>cc: @infomalang @infokostmalang @infomalangraya @Kontrakan_MLG https://t.co/em54FuTt0B |
| 3 | 1 | kost PUTRA 276 di jl.ikan piranha atas, gg. 20 no.276<br>an. ibu atik/p febri<br>KM dlm 700k/bln or 8.4 jt/th<br>KM Luar 500k/bln or 5.4 jt/th<br>Harga sudah termasuk air, PLN dan WIfi<br>fasitas :<br>dapur, bebas jam malam, bawa kunci sendiri<br>cp whatasp: 083834647466 https://t.co/hctNz2z0hM |
| 4 | 1 | Kost putra Jln.Tlogo Agung nomor 45i.<br>Kosong 2 kamar.<br>Harga : 2,4jt/semester<br>Free listrik air.<br>Tiap anak dibawak kunci pagar dn garasi.<br>Hub Bu Rini 085730523404. @pijarwirastani https://t.co/FrbFMjbyE3 |
| 5 | 2 | Kost mhsw putri tahunan<br>Jl.Kesumba dalam 19 |

| | | |
|---|---|---|
| | | -Dkt UB Poltek jalan kaki 5 mnt<br>-Parkir motor luas<br>-Dapat kunci pagar<br>- free Wifi<br>- Kamar lengkap<br>-Dapur, kompor gas<br>-jemuran di lt. 3<br>-Ada pembantu kos untuk bersih2<br>WA: 081231419210/ 081366253796 @infokostmalang https://t.co/Jer89pVpjq |
| 6 | 2 | Kost putri jl.mayjen panjaitan dalam gang 19 no.49<br>Jln kaki 3 mnt ke UB<br>Harga mulai 4jt - 6jt/th<br>Fasilitas brsama :<br>Tv, Wifi, Parkir Motor, Dapur di lantai 2 dan 3(ada kompor dan gas)<br>Air PDAM<br>CCTV 24 jam*<br>Tempat menjemur pakaian<br>Wa : 085259240800 https://t.co/odRXT9XYzo |
| 7 | ? | Cari rumah untuk dikontrakkan? |

**Keterangan:**

No 1-6: Data Latih No 7: Data Uji

**Tahap *Preprocessing***

**Langkah 2.** Melakukan proses *Case Folding*

Tahap *case folding* ini dilakukan untuk melakukan proses mengubah huruf kapital menjadi huruf kecil. Hasil proses *case folding* dapat dilihat pada Tabel 4.2 dan Tabel 4.3:

**Tabel 4. 2 Hasil *Case Folding* Data Latih**

| No | Label | Tweet |
|---|---|---|
| 1 | 0 | dikontrakkan<br>rumah baru dekat kampus universitas brawijaya,polinema,unisma,um,jalan soehatt<br>(jl. dewandaru dalam 37 c)<br><br>monggo hub. ☎ 085649691086 (wa) |

| | | |
|---|---|---|
| | | #infokostmalang #infomalang<br>cc @infokostmalang @infomalangraya @pojokpolinema @infomalang<br><br>#infokostmalang #dekatkampus https://t.co/8mkxwdto53 |
| 2 | 0 | dikontrakkan rumah di daerah tidar, 6 kamar tidur, 1 kamar mandi, jemuran di atas. dekat ke stiki, ub, um, uin, itn..<br>monggo yg minat bisa hub 081259809311<br><br>cc: @infomalang @infokostmalang @infomalangraya @kontrakan_mlg https://t.co/em54futt0b |
| 3 | 1 | kost putra 276 di jl.ikan piranha atas, gg. 20 no.276<br>an. ibu atik/p febri<br>km dlm 700k/bln or 8.4 jt/th<br>km luar 500k/bln or 5.4 jt/th<br>harga sudah termasuk air, pln dan wifi<br>fasitas :<br>dapur, bebas jam malam, bawa kunci sendiri<br>cp whatasp: 083834647466 https://t.co/hctnz2z0hm |
| 4 | 1 | kost putra jln.tlogo agung nomor 45i.<br>kosong 2 kamar.<br>harga : 2,4jt/semester<br>free listrik air.<br>tiap anak dibawak kunci pagar dn garasi.<br>hub bu rini 085730523404. @pijarwirastani https://t.co/frbfmjbye3 |
| 5 | 2 | kost mhsw putri tahunan<br>jl.kesumba dalam 19<br>-dkt ub poltek jalan kaki 5 mnt<br>-parkir motor luas<br>-dapat kunci pagar<br>- free wifi<br>- kamar lengkap<br>-dapur, kompor gas<br>-jemuran di lt. 3<br>-ada pembantu kos untuk bersih2<br>wa: 081231419210/ 081366253796 @infokostmalang https://t.co/jer89pvpjq |
| 6 | 2 | kost putri jl.mayjen panjaitan dalam gang 19 no.49<br>jln kaki 3 mnt ke ub |

| | | |
|---|---|---|
| | | harga mulai 4jt - 6jt/th<br>fasilitas brsama :<br>tv, wifi, parkir motor, dapur di lantai 2 dan 3(ada kompor dan gas)<br>air pdam<br>cctv 24 jam*<br>tempat menjemur pakaian<br>wa : 085259240800 https://t.co/odrxt9xyzo |

**Tabel 4.3 Hasil *Case Folding* Data Uji**

| | | |
|---|---|---|
| 7 | ? | cari rumah untuk dikontrakkan? |

**Langkah 3.** Melakukan *cleaning* terhadap *tweet* untuk menghilangkan karakter dan pola karakter yang tidak diperlukan.

Pada tahap ini masing-masing data dari dataset akan dilakukan *cleaning* dan satu-persatu, yang berguna untuk menghilangkan karakter dan pola karakter seperti URL, *hashtag*, dan nama pengguna. Hasil *Cleaning* dapat dilihat pada Tabel 4.4 dan 4.5

**Tabel 4.4 Hasil *Cleaning* Data Latih**

| No | Label | Tweet |
|---|---|---|
| 1 | 0 | dikontrakkan rumah baru dekat kampus universitas brawijaya polinema unisma um jalan soehatt jl dewandaru dalam 37 c monggo hub 085649691086 wa cc |
| 2 | 0 | dikontrakkan rumah di daerah tidar 6 kamar tidur 1 kamar mandi jemuran di atas dekat ke stiki ub um uin itn monggo yg minat bisa hub 081259809311 cc |
| 3 | 1 | kost putra 276 di jl ikan piranha atas gg 20 no 276 an ibu atik p febri km dlm 700k bln or 8 4 jt th km luar 500k bln or 5 4 jt th harga sudah termasuk air pln dan wifi fasitas dapur bebas jam malam bawa kunci sendiri cp whatasp 083834647466 |
| 4 | 1 | kost putra jln tlogo agung nomor 45i kosong 2 kamar harga 2 4jt semester free listrik air tiap anak dibawak kunci pagar dn garasi hub bu rini 085730523404 |
| 5 | 2 | kost mhsw putri tahunan jl kesumba dalam 19 dkt ub poltek jalan kaki 5 mnt parkir motor luas dapat kunci pagar free wifi kamar lengkap dapur kompor gas jemuran di lt 3 ada pembantu kos untuk bersih2 wa 081231419210 081366253796 |
| 6 | 2 | kost putri jl mayjen panjaitan dalam gang 19 no 49 jln kaki 3 mnt ke ub harga mulai 4jt 6jt th fasilitas brsama tv wifi parkir motor dapur di lantai 2 dan 3 ada kompor dan gas air pdam cctv 24 jam tempat menjemur pakaian wa 085259240800 |

**Tabel 4.5 Hasil *Cleaning* Data Uji**

| | | |
|---|---|---|
| 7 | ? | cari rumah untuk dikontrakkan |

**Langkah 4.** Melakukan proses *Filtering*

Proses Filtering atau bisa di sebut *Stopword Removal* adalah proses membuang kata-kata yang tidak penting yang tidak diperlukan, pada proses ini sistem menggunakan *Filtering* dari API pujangga inaNLP. Berikut ini adalah contoh proses *Stopword Removal* yang disajikan pada Tabel 4.6 dan 4.7.

**Tabel 4.6 Hasil *Filtering* Data Latih**

| No | Label | Tweet |
|---|---|---|
| 1 | 0 | dikontrakkan rumah kampus universitas brawijaya polinema unisma um jalan soehatt jl dewandaru 37 c monggo hub 085649691086 wa cc |
| 2 | 0 | dikontrakkan rumah daerah tidar 6 kamar tidur 1 kamar mandi jemuran stiki ub um uin itn monggo yg minat hub 081259809311 cc |
| 3 | 1 | kost putra 276 di jl ikan piranha atas gg 20 no 276 an ibu atik p febri km dlm 700k bln or 8 4 jt th km luar 500k bln or 5 4 jt th harga sudah termasuk air pln dan wifi fasitas dapur bebas jam malam bawa kunci sendiri cp whatasp 083834647466 |
| 4 | 1 | kost putra jln tlogo agung nomor 45i kosong 2 kamar harga 2 4jt semester free listrik air anak dibawak kunci pagar dn garasi hub bu rini 085730523404 |
| 5 | 2 | kost mhsw putri tahunan jl kesumba 19 dkt ub poltek jalan kaki 5 mnt parkir motor luas kunci pagar free wifi kamar lengkap dapur kompor gas jemuran lt 3 pembantu kos bersih2 wa 081231419210 081366253796 |
| 6 | 2 | kost putri jl mayjen panjaitan gang 19 no 49 jln kaki 3 mnt ub harga 4jt 6jt th fasilitas brsama tv wifi parkir motor dapur lantai 2 3 kompor gas air pdam cctv 24 jam menjemur pakaian wa 085259240800 |

**Tabel 4.7 Hasil *Filtering* Data Uji**

| | | |
|---|---|---|
| 7 | ? | cari rumah dikontrakkan |

**Langkah 5.** Melakukan proses *Stemming*

Pada langkah kelima adalah proses *stemming,* proses ini berfungsi untuk mengubah kata yang sudah dilakukan *filtering* menjadi kata dasar. Dibawah ini adalah contoh proses Stemming pada Tabel 4.8:

**Tabel 4.8 Contoh Proses *Stemming***

| Hasil *Filtering* | Hasil *Stemming* |
|---|---|
| mencari | Cari |
| dikontrakkan | Kontrak |
| tersedia | Sedia |

Untuk hasil lengkap dari proses *stemming* untuk data latih dan data uji telah disajikan pada Tabel 4.9 dan Tabel 4.10:

**Tabel 4.9 Hasil *Stemming* Data Latih**

| No | Label | Tweet |
|---|---|---|
| 1 | 0 | kontrak rumah kampus universitas brawijaya polinema unisma um jalan soehatt jl dewandaru 37 c monggo hub 085649691086 wa cc |
| 2 | 0 | kontrak rumah daerah tidar 6 kamar tidur 1 kamar mandi jemur stiki ub um uin itn monggo yg minat hub 081259809311 cc |

| | | |
|---|---|---|
| 3 | 1 | kost putra 276 jl ikan piranha gg 20 no 276 an atik p febri km dlm 700k bln or 8 4 jt th km 500k bln or 5 4 jt th harga air pln wifi fasitas dapur bebas jam malam bawa kunci cp whatasp 083834647466 |
| 4 | 1 | kost putra jln tlogo agung nomor 45i kosong 2 kamar harga 2 4jt semester free listrik air anak bawak kunci pagar dn garasi hub bu rini 085730523404 |
| 5 | 2 | kost mhsw putri tahun jl kesumba 19 dkt ub poltek jalan kaki 5 mnt parkir motor luas kunci pagar free wifi kamar lengkap dapur kompor gas jemur lt 3 bantu kos bersih2 wa 081231419210 081366253796 |
| 6 | 2 | kost putri jl mayjen panjaitan gang 19 no 49 jln kaki 3 mnt ub harga 4jt 6jt th fasilitas brsama tv wifi parkir motor dapur lantai 2 3 kompor gas air pdam cctv 24 jam jemur pakai wa 085259240800 |

**Tabel 4.10 Hasil *Filtering* Data Uji**

| | | |
|---|---|---|
| 7 | ? | cari rumah kontrak |

**Langkah 6.** Melakukan proses *Tokenizing*

Pada langkah terakhir *preprocessing* adalah proses *tokenizing,* proses ini berfungsi untuk memecah sebuah kalimat menjadi kata hasil *tokenizing* dapat dilihat 4.11 dan 4.12:

**Tabel 4.11 Hasil *Tokenizing* Data Latih**

| No | Label | Tweet |
|---|---|---|
| 1 | 0 | kontrak \| rumah \| kampus \| universitas \| brawijaya \| polinema \| unisma \| um \| jalan \| soehatt \| jl \| dewandaru \| 37 \| c \| monggo \| hub \| 085649691086 \| wa \| cc |
| 2 | 0 | kontrak \| rumah \| daerah \| tidar \| 6 \| kamar \| tidur \| 1 \| kamar \| mandi \| jemur \| stiki \| ub \| um \| uin \| itn \| monggo \| yg \| minat \| hub \| 081259809311 \| cc |
| 3 | 1 | kost \| putra \| 276 \| jl \| ikan \| piranha \| gg \| 20 \| no \| 276 \| an \| atik \| p \| febri \| km \| dlm \| 700k \| bln \| or \| 8 \| 4 \| jt \| th \| km \| 500k \| bln \| or \| 5 \| 4 \| jt \| th \| harga \| air \| pln \| wifi \| fasitas \| dapur \| bebas \| jam \| malam \| bawa \| kunci \| cp \| whatasp \| 083834647466 |
| 4 | 1 | kost \| putra \| jln \| tlogo \| agung \| nomor \| 45i \| kosong \| 2 \| kamar \| harga \| 2 \| 4jt \| semester \| free \| listrik \| air \| anak \| bawak \| kunci \| pagar \| dn \| garasi \| hub \| bu \| rini \| 085730523404 |
| 5 | 2 | kost \| mhsw \| putri \| tahun \| jl \| kesumba \| 19 \| dkt \| ub \| poltek \| jalan \| kaki \| 5 \| mnt \| parkir \| motor \| luas \| kunci \| pagar \| free \| wifi \| kamar \| lengkap \| dapur \| kompor \| gas \| jemur \| lt \| 3 \| bantu \| kos \| bersih2 \| wa \| 081231419210 \| 081366253796 |
| 6 | 2 | kost \| putri \| jl \| mayjen \| panjaitan \| gang \| 19 \| no \| 49 \| jln \| kaki \| 3 \| mnt \| ub \| harga \| 4jt \| 6jt \| th \| fasilitas \| brsama \| tv \| wifi \| parkir \| motor \| dapur \| lantai \| 2 \| 3 \| kompor \| gas \| air \| pdam \| cctv \| 24 \| jam \| jemur \| pakai \| wa \| 085259240800 |

**Tabel 4.12 Hasil *Tokenizing* Data Uji**

| | | |
|---|---|---|
| 7 | ? | cari \| rumah \| kontrak |

**Tahap Proses Seleksi Fitur**

**Langkah 6.** Melakukan proses yang dilakukan untuk memilih fitur-fitur mana saja yang lebih merepresentasikan tiap kelasnya. Seleksi yang dilakukan akan mengakibatkan berkurangnya fitur yang digunakan ketika melakukan klasifikasi, pada penelitian ini metode seleksi fitur yang digunakan adalah metode IG, pada proses ini akan diberikan contoh perhitungan menggunakan term kontrak, kemunculan term kontrak pada dokumen dan kelas tertentu dapat dilihat pada Tabel 4.13

**Tabel 4.13 Frekuensi kemunculan kata**

| Term | Frekuensi Term | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Dok 1 | Dok 2 | Dok 3 | Dok 4 | Dok 5 | Dok 6 |
| Kontrak | 1 | 1 | 0 | 0 | 0 | 0 |

Data di atas akan digunakan untuk menghitung nilai IG dari term kontrak menggunakan rumus yang dapat dilihat pada Persamaan 4.1.

$$= -(0{,}477121255) + (0) + (-0{,}200686664)$$

$$= 0{,}276434591$$

Hasil perhitungan IG seluruh term pada Data Latih dapat dilihat pada Tabel 4.14

**Tabel 4.14 Hasil Perhitungan *Information Gain***

| | | Kontrakan | | Kost Putra | | Kost Putri | | |
|---|---|---|---|---|---|---|---|---|
| No | Term | Dok 1 | Dok 2 | Dok 3 | Dok 4 | Dok 5 | Dok 6 | IG |
| 1 | kontrak | 1 | 1 | 0 | 0 | 0 | 0 | 0,276435 |
| 2 | rumah | 1 | 1 | 0 | 0 | 0 | 0 | 0,276435 |
| 3 | kampus | 1 | 0 | 0 | 0 | 0 | 0 | 0,095333 |
| 4 | universitas | 1 | 0 | 0 | 0 | 0 | 0 | 0,095333 |
| 5 | brawijaya | 1 | 0 | 0 | 0 | 0 | 0 | 0,095333 |
| 6 | polinema | 1 | 0 | 0 | 0 | 0 | 0 | 0,095333 |
| 7 | unisma | 1 | 0 | 0 | 0 | 0 | 0 | 0,095333 |
| 8 | um | 1 | 1 | 0 | 0 | 0 | 0 | 0,276435 |
| 9 | jalan | 1 | 0 | 0 | 0 | 1 | 0 | 0,075748 |
| 10 | soehatt | 1 | 0 | 0 | 0 | 0 | 0 | 0,095333 |
| 11 | jl | 1 | 0 | 1 | 0 | 1 | 1 | 0,075748 |
| 12 | dewandaru | 1 | 0 | 0 | 0 | 0 | 0 | 0,095333 |
| 13 | 37 | 1 | 0 | 0 | 0 | 0 | 0 | 0,095333 |
| 14 | c | 1 | 0 | 0 | 0 | 0 | 0 | 0,095333 |
| 15 | monggo | 1 | 1 | 0 | 0 | 0 | 0 | 0,276435 |
| 16 | hub | 1 | 1 | 0 | 1 | 0 | 0 | 0,200687 |

**Tabel 4.15 Hasil Perhitungan *Information Gain* (lanjutan)**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 17 | 085649691086 | 1 | 0 | 0 | 0 | 0 | 0 | 0,095333 |
| 18 | wa | 1 | 0 | 0 | 0 | 1 | 1 | 0,200687 |
| … | …. | … | … | … | … | … | … | … |
| 126 | 085259240800 | 0 | 0 | 0 | 0 | 0 | 1 | 0,095333 |

Setelah dilakukan perhitungan IG term, term akan diurutkan berdasarkan nilai IG tertinggi hingga terendah dan akan dipilih sebanyak 50 % dari 128 sehingga yang diambil hanya 63 fitur yang dapat dilihat pada Tabel 4.16:

**Tabel 4.16 Pengurutan dan Seleksi Fitur**

| No | Term | IG(t) |
|---|---|---|
| 1 | kontrak | 0,276435 |
| 2 | rumah | 0,276435 |
| 3 | um | 0,276435 |
| 4 | monggo | 0,276435 |
| 5 | cc | 0,276435 |
| 6 | kost | 0,276435 |
| 7 | putra | 0,276435 |
| 8 | putri | 0,276435 |
| 9 | 19 | 0,276435 |
| 10 | kaki | 0,276435 |
| 11 | mnt | 0,276435 |
| 12 | parkir | 0,276435 |
| 13 | motor | 0,276435 |
| 14 | kompor | 0,276435 |
| 15 | gas | 0,276435 |
| 16 | 3 | 0,276435 |
| 17 | hub | 0,200687 |
| 18 | wa | 0,200687 |
| 19 | jemur | 0,200687 |
| 20 | ub | 0,200687 |
| 21 | harga | 0,200687 |
| 22 | air | 0,200687 |
| 23 | wifi | 0,200687 |
| 24 | dapur | 0,200687 |
| 25 | kunci | 0,200687 |
| 26 | mhsw | 0,095333 |
| 27 | tahun | 0,095333 |
| 28 | kesumba | 0,095333 |
| 29 | dkt | 0,095333 |
| 30 | poltek | 0,095333 |
| 31 | luas | 0,095333 |

**Tabel 4.17 Pengurutan dan Seleksi Fitur (lanjutan)**

| | | |
|---|---|---|
| 32 | lengkap | 0,095333 |
| 33 | lt | 0,095333 |
| 34 | bantu | 0,095333 |
| 35 | kos | 0,095333 |
| 36 | bersih2 | 0,095333 |
| 37 | 081231419210 | 0,095333 |
| 38 | 081366253796 | 0,095333 |
| 39 | mayjen | 0,095333 |
| 40 | panjaitan | 0,095333 |
| 41 | gang | 0,095333 |
| 42 | 49 | 0,095333 |
| 43 | 6jt | 0,095333 |
| 44 | fasilitas | 0,095333 |
| 45 | brsama | 0,095333 |
| 46 | tv | 0,095333 |
| 47 | lantai | 0,095333 |
| 48 | pdam | 0,095333 |
| 49 | cctv | 0,095333 |
| 50 | 24 | 0,095333 |
| 51 | pakai | 0,095333 |
| 52 | 085259240800 | 0,095333 |
| 53 | kampus | 0,095333 |
| 54 | universitas | 0,095333 |
| 55 | brawijaya | 0,095333 |
| 56 | polinema | 0,095333 |
| 57 | unisma | 0,095333 |
| 58 | soehatt | 0,095333 |
| 59 | dewandaru | 0,095333 |
| 60 | 37 | 0,095333 |
| 61 | c | 0,095333 |
| 62 | 085649691086 | 0,095333 |

**Langkah 7.** Melakukan proses pembobotan kata

Pada pembobotan kata dilakukan untuk perhitungan klasifikasi menggunakan Naive Bayes. Pada tahap ini dilakukan pembobotan kata dengan menggunakan metode TF-IDF-CF dengan menggunakan rumus yang dapat dilihat pada Persamaan 2.1.

Yang pertama dilakukan adalah menghitung frekuensi kemunculan sebuah term pada setiap dokumen(TF) dan menghitung jumlah dokumen yang terdapat term tersebut(DF), perhitungannya dapat dilihat pada Tabel 4.18

**Tabel 4.18 Perhitungan TF dan DF**

| Term | Dok 1 | Dok 2 | Dok 3 | Dok 4 | Dok 5 | Dok 6 | df |
|---|---|---|---|---|---|---|---|
| kontrak | 1 | 1 | 0 | 0 | 0 | 0 | 2 |
| rumah | 1 | 1 | 0 | 0 | 0 | 0 | 2 |
| um | 1 | 1 | 0 | 0 | 0 | 0 | 2 |
| monggo | 1 | 1 | 0 | 0 | 0 | 0 | 2 |
| cc | 1 | 1 | 0 | 0 | 0 | 0 | 2 |
| kost | 0 | 0 | 1 | 1 | 1 | 1 | 4 |
| putra | 0 | 0 | 1 | 1 | 0 | 0 | 2 |
| putri | 0 | 0 | 0 | 0 | 1 | 1 | 2 |
| 19 | 0 | 0 | 0 | 0 | 1 | 1 | 2 |
| kaki | 0 | 0 | 0 | 0 | 1 | 1 | 2 |
| mnt | 0 | 0 | 0 | 0 | 1 | 1 | 2 |
| parkir | 0 | 0 | 0 | 0 | 1 | 1 | 2 |
| … | … | … | … | … | … | … | … |
| 085649691086 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |

Kemudian menghitung kemunculan term pada dokumen di kelas(CF) tertentu yang dapat dilihat pada Tabel 4.19:

**Tabel 4.19 Perhitungan CF**

| Term | cf kelas 0 | cf kelas 1 | cf kelas 2 |
|---|---|---|---|
| kontrak | 2 | 0 | 0 |
| rumah | 2 | 0 | 0 |
| um | 2 | 0 | 0 |
| monggo | 2 | 0 | 0 |
| cc | 2 | 0 | 0 |
| kost | 0 | 2 | 2 |
| putra | 0 | 2 | 0 |

**Tabel 4.20 Perhitungan CF (lanjutan)**

| | | | |
|---|---|---|---|
| putri | 0 | 0 | 2 |
| 19 | 0 | 0 | 2 |
| kaki | 0 | 0 | 2 |
| mnt | 0 | 0 | 2 |
| parkir | 0 | 0 | 2 |
| … | … | … | … |
| 085649691086 | 1 | 0 | 0 |

Setelah diketahui TF,DF, dan CF kemudian akan dihitung menggunakan Persamaan 4.2, contoh perhitungan menggunakan term kontrak dokumen 1 pada kelas kontrakan dapat dilihat seperti berikut:

$$w_{(kontrak)} = \log(1+1) * log\left(\frac{6+1}{2}\right) * \frac{2}{2}$$
$$= (0{,}301029996) * (0{,}544068044) * 1$$
$$= 0{,}163780801$$

Berikut ini adalah hasil perhitungan TF-IDF-CF pada seluruh term di setiap dokumen dapat dilihat pada Tabel 4.21

**Tabel 4.21 Perhitungan TD-IDF-CF**

| | Kontrakan | | Kost Putra | | Kost Putri | |
|---|---|---|---|---|---|---|
| TF-IDF-CF | Dok 1 | Dok 2 | Dok 3 | Dok 4 | Dok 5 | Dok 6 |
| kontrak | 0,163781 | 0,163781 | 0 | 0 | 0 | 0 |
| rumah | 0,163781 | 0,163781 | 0 | 0 | 0 | 0 |
| um | 0,163781 | 0,163781 | 0 | 0 | 0 | 0 |
| monggo | 0,163781 | 0,163781 | 0 | 0 | 0 | 0 |
| cc | 0,163781 | 0,163781 | 0 | 0 | 0 | 0 |
| kost | 0 | 0 | 0,073162 | 0,073162 | 0,073162 | 0,073162 |
| putra | 0 | 0 | 0,163781 | 0,163781 | 0 | 0 |
| putri | 0 | 0 | 0 | 0 | 0,163781 | 0,163781 |
| 19 | 0 | 0 | 0 | 0 | 0,163781 | 0,163781 |
| kaki | 0 | 0 | 0 | 0 | 0,163781 | 0,163781 |
| mnt | 0 | 0 | 0 | 0 | 0,163781 | 0,163781 |
| parkir | 0 | 0 | 0 | 0 | 0,163781 | 0,163781 |

**Tabel 4.22 Perhitungan TD-IDF-CF**

| | | | | | | |
|---|---|---|---|---|---|---|
| … | … | … | … | … | … | … |
| 085649691086 | 0,1272 | 0 | 0 | 0 | 0 | 0 |

**Langkah 8.** Melakukan proses klasifikasi *Naïve Bayes*

Setelah diketahui jumlah frekuensi masing-masing kata uji pada dokumen latih maka dilakukan perhitungan *prior*. Tabel perhitungan *prior* ditunjukkan pada Tabel 4.23.

**Tabel 4.23 Perhitungan *Prior***

| P(Kontrakan) | P(Kost Putra) | P(Kost Putri) |
|---|---|---|
| $\frac{2}{6} = 0{,}33$ | $\frac{2}{6} = 0{,}33$ | $\frac{2}{6} = 0{,}33$ |

Setelah didapatkan hasil dari prior kontrakan, kost putra dan kost putri maka langkah selanjutnya adalah menghitung *conditional probability* dengan menggunakan yang ditunjukkan pada Tabel 4.24.

**Tabel 4.24 Perhitungan *Conditional Probability***

| Kata | Conditional Probability (Kontrakan) |
|---|---|
| rumah | $\frac{1}{\sqrt{2\pi}\,(0{,}316)} e^{-\frac{(0{,}163780801-0{,}163780801)^2}{2(0.1)^2}} = 1{,}2615$ |
| kontrak | $\frac{1}{\sqrt{2\pi}\,(0{,}316)} e^{-\frac{(0{,}163780801-0{,}163780801)^2}{2(0.1)^2}} = 1{,}2615$ |
| Kata | Conditional Probability (Kost Putra) |
| rumah | 1 |
| kontrak | 1 |
| Kata | Conditional Probability (Kost Putri) |
| rumah | 1 |
| kontrak | 1 |

Setelah didapatkan nilai dari *conditional probability* pada tiap kata masing-masing kelas, maka langkah selanjutnya yaitu menghitung *total conditional*

*probability* dengan cara mengalikan hasil dari setiap kata pada masing-masing kelas. Perhitungan *total conditional probability* ditunjukkan pada Tabel 4.25.

**Tabel 4.25 Perhitungan Total *Conditional Probability***

| Total Conditional Probability (Kontrakan) |
|---|
| 1.5915494309189537 |
| Total Conditional Probability (Kost Putra) |
| 1 |
| Total Conditional Probability (Kost Putri) |
| 1 |

Kemudian setelah didapatkan nilai dari *total conditional probability* maka langkah selanjutnya adalah menghitung *posterior*. Rumus untuk menghitung *posterior yaitu prior * total conditional probability*. Dan hasil dari perkalian tersebut dilakukan perbandingan untuk mencari nilai tertinggi. Nilai tertinggi tersebut akan menentukan sebuah dokumen masuk ke dalam kelas sentimen positif atau negatif. Perhitungan *posterior* ditunjukkan pada Tabel 4.26.

**Tabel 4.26 Perhitungan *Posterior***

| Posterior (Kontrakan) | Kelas |
|---|---|
| 0.5305164769729845 | Kontrakan |
| 0,3333333 | Kost Putra |
| 0,3333333 | Kost Putri |

Berdasarkan hasil dari perhitungan Tabel 4.26 dapat disimpulkan bahwa nilai *posterior* pada kelas Kontrakan memiliki nilai yang lebih tinggi dibandingkan dengan nilai posterior pada kelas Kontrakan dan kelas Kost Putra. Sehingga data uji pada no 4 termasuk dalam kelas Kontrakan.

| No | Label | Tweet |
|---|---|---|
| 7 | 0 | Cari rumah untuk dikontrakkan? |

## 4.5 Perancangan Pengujian

Pada sub bab ini penelitian akan membahas tentang perancangan pada pengujian data terhadap sistem yang telah diimplementasikan. Untuk pengujian masing-masing kelas akan diambil 20 data uji sehingga totalnya adalah 60 data uji dan akan dijalankan oleh sistem, kemudian akan didapatkan hasil akurasi sistem yang digunakan untuk mengukur apakah sistem mampu mengklasifikasikan *tweet* tempat tinggal berdasarkan kategorinya masing-masing dengan baik. Dalam penelitian ini dilakukan 4 kali pengujian untuk mengetahui pengaruh pembobotan TF-IDF-CF dan seleksi fitur IG, yang pertama adalah menggunakan TF-IDF-CF dan IG yang kedua dengan TF-IDF-CF dan tanpa IG, yang ketiga tanpa TF-IDF-CF dan dengan IG, dan yang terakhir tanpa TF-IDF-CF dan tanpa IG, sehingga nanti akan didapatkan 4 akurasi yang berbeda yang dapat digunakan untuk membandingkan dan menganalisis kinerja dari pembobotan kata dan seleksi fitur yang digunakan. Untuk perancangan pengujian sendiri dapat dilihat pada tabel 4.27

**Tabel 4.27 Perancangan Pengujian**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | | | | | | |

Tabel di atas akan digunakan untuk setiap pengujian yang dilakukan yaitu sebanyak 4 kali yang dimana nanti hasil akurasi akan didapatkan dengan menghitung jumlah klasifikasi yang benar dibagi dengan jumlah data uji dikali dengan 100%. Sehingga nanti dapat diambil dilihat kombinasi metode mana yang memiliki nilai akurasi tertinggi.

# BAB 5 IMPLEMENTASI

## 5.1 Perangkat Keras

Dalam mengimplementasikan sistem yang dibuat perangkat keras yang dipakai adalah laptop ASUS A456UF dengan spesifikasi:

1. Processor Intel(R) Core(TM) i5-6200 CPU @2.30GHz (4 CPUs), ~2.8GHz.
2. Harddisk 500 GB.
3. Memori RAM 4 GB.

## 5.2 Perangkat Lunak

Sedangkan untuk perangkat lunak yang dipakai untuk implementasi sistem yang dibangun adalah sebagai berikut ini:

1. Sistem Operasi yang digunakan adalah Windows 10 Pro 64-bit.
2. Editor yang digunakan adalah Spyder Anaconda Python 3.6.

## 5.3 Batasan Implementasi

Batasan-batasan yang dipakai dalam implementasi sistem pada penelitian ini adalah sebagai berikut:

1. Implementasi sistem pada penelitian ini memakai bahasa pemrograman python.
2. Data disimpan kedalam format *excel* yang digunakan untuk proses implementasi.
3. Metode yang digunakan dalam klasifikasi tempat tinggal di kota Malang adalah *Naïve Bayes*.
4. Data disini berasal dari pengumpulan data dari API akun Twitter @infomalang dan @infokostmalang.
5. Kelas data terdiri dari kost putra, kost putri, dan kontrakan.

## 5.4 Implementasi Algoritme

Pada implementasi algoritme disini menjelaskan terkait kode program dari sistem yang mengacu pada perancangan proses pada bab sebelumnya. Dimana meliputi *preprocessing*, seleksi fitur, dan klasifikasi menggunakan metode Naïve Bayes.

### 5.4.1 Implementasi *Preprocessing*

Pada implementasi *preprocessing* dilakukan untuk mempersiapkan data sebelum diolah, data disini berupa teks. Data yang siap akan diproses dalam metode NBC. Pada *preprocessing* ini terdapat beberapa tahapan, yaitu *case folding, cleaning*, *filtering*, *stemming, dan tokenizing*. Tiap tahapan mempunyai

tujuan masing-masing, yang pertama ada *case folding* tahap ini mengubah seluruh huruf pada teks menjadi huruf kecil, kemudian ada *cleaning* yaitu proses untuk menghilangkan karakter/simbol dan pola karakter yang biasanya tidak diperlukan contoh karakter adalah tanda baca, sedangkan pola karakter seperti URL , yang ketiga ada *filtering* atau biasa disebut *stopword removal* proses ini berguna untuk menghilangkan kata-kata yang tidak penting seperti dan, atau, dll, selanjutnya adalah *stemming* yaitu mengubah kata menjadi kata dasar, dan yang terakhir adalah proses *tokenizing* proses ini berguna untuk memecah kalimat menjadi kata atau biasa disebut token, pada kasus ini kenapa *tokenizing* dilakukan di akhir karena pada sistem yang dibuat untuk melakukan proses *stemming*, dan *filtering*, sistem menggunakan sebuah API yaitu API pujangga inaNLP sehingga *tokenizing* dilakukan terakhir untuk mengurangi waktu komputasi, karena jika *tokenizing* dilakukan di awal maka API akan digunakan sebanyak token yang ada yang tentu akan memperlambat proses komputasi. Implementasi proses *Preprocessing* disajikan pada potongan Source Code 5.1.

```
1.  import re
2.  import requests
3.
4.  def stopword(t):
5.      r = requests.post('http://127.0.0.1:9000/stopwords',
6.  json={'string': t})
7.      return r.text[28:-2]
8.
9.  def stemmer(t):
10.     r = requests.post('http://127.0.0.1:9000/stemmer',
11. json={'string': t})
12.     return r.text[28:-2]
13.
14. def hapussimbol(t):
15.     z = re.sub(r'#[\w]*|@[\w]*', ' ', t)
16.     n = re.sub(r'[^\w]', ' ', z)
17.     b = re.sub(r'(http?)(.*)', ' ', n)
18.     m = re.sub(r'(\s{1,}|(uD.*))', ' ', b)
19.     return m
20.
21. def preprot(tx):
22.     teks = tx.casefold()
23.     teks = hapussimbol(teks)
24.     teks = stopword(teks)
25.     teks = stemmer(teks)
26.     teks = teks.split()
27.     return teks
```

**Source Code 5.1 Implementasi Proses Preprocessing**

Penjelasan Source Code 5.1 Proses *Preprocessing*, baris ke:

1. Mengimport library *re* yang berguna sebagai *regex*.
2. Mengimport library *request* yang berfungsi untuk memanggil *API*.
4-7. Fungsi untuk menghilangkan *stopword* fungsi ini menerima parameter berupa *string* yang kemudian akan di proses melalui *API* dan akan mengembalikan data berupa teks yang sudah dihilangkan *stopword*-nya.

9-12. Fungsi untuk stemming fungsi ini menerima parameter berupa *string* yang kemudian akan di proses melalui *API* dan akan mengembalikan data berupa teks yang sudah diubah menjadi kata dasar.

14-19. Fungsi untuk *cleaning*, fungsi ini menerima parameter berupa *string* yang kemudian akan di proses untuk menghilangkan simbol, URL, dan pola karakter yang tidak diperlukan.

21-27. Fungsi untuk melakukan tahap *preprocessing* dengan parameter berupa *string*.

### 5.4.2 Implementasi Algoritme Seleksi Fitur

Algoritme seleksi fitur digunakan untuk memilih fitur-fitur yang digunakan dalam proses klasifikasi, untuk metode yang digunakan adalah IG, metode ini akan menghitung kata/term dengan nilai IG tertinggi yang akan diurutkan dari terbesar hingga terkecil, kemudian akan diseleksi dengan jumlah tertentu, kata yang terpilih adalah kata dengan nilai IG teratas karena nilai IG yang tinggi memiliki arti bahwa sebuah kata dapat merepresentasikan suatu kelas.

```
1.  def hitungpt(kelas):
2.      nilai = {}
3.      #IG = []
4.      for x in dokumenu:
5.          if(x[1]==kelas):
6.              for y in x[0]:
7.                  for z in term:
8.                      if(str(y)==str(z) and
9.  str(nilai.get(z))=='None'):
10.                         nilai.update({z:1})
11.                     elif(str(y)==str(z)):
12.                         nilai[z]=nilai.get(z)+1
13.     return(nilai)
14.
15. for x in range(3):
16.     print("==========================================")
17.     for y in term:
18.         #y = "kontrak"
19.         #print(term.get(y))
20.         #print(str(IG[x][0].get(y)))
21.         ppt = term.get(y)/len(dokumen)
22.         ppnt = (len(dokumen)-term.get(y))/len(dokumen)
23.         if(str(IG[x][0].get(y))=='None'):
24.             pt = 0
25.             pnt = ((lenkelas-0)/(len(dokumenu)-
26. term.get(y)))*(math.log10(((lenkelas-0)/(len(dokumenu)-
27. term.get(y)))))
28.             print(y+"\t"+str(pt)+" "+str(pnt))
29.             #print(y+"\t"+str(ppt)+" "+str(ppnt))
30.             dictpt.update({y:pt})
31.             dictpnt.update({y:pnt})
32.             totpt.update({y:ppt})
33.             totpnt.update({y:ppnt})
34.         else:
```

```
35.             pt =
36. (float(IG[x][0].get(y))/term.get(y))*(math.log10((float(IG[
37. x][0].get(y))/term.get(y))))
38.             if((2-float(IG[x][0].get(y)))==0):
39.                 pnt = 0
40.             else:
41.                 pnt = ((lenkelas-
42. float(IG[x][0].get(y)))/(len(dokumenu)-
43. term.get(y)))*(math.log10(((lenkelas-
44. float(IG[x][0].get(y)))/(len(dokumenu)-term.get(y)))))
45.             print(y+"\t"+str(pt)+" "+str(pnt))
46.             #print(y+"\t"+str(ppt)+" "+str(ppnt)+"= - = - =
47. - = - = - = - = - =")
48.             dictpt.update({y:pt})
49.             dictpnt.update({y:pnt})
50.             totpt.update({y:ppt})
51.             totpnt.update({y:ppnt})
52.     totIGpt.append([dictpt,x])
53.     totIGpnt.append([dictpnt,x])
54.     dictpt={}
55.     dictpnt={}
56.
57. for x in term:
58.     for y in range(3):
59.         totalt += (totIGpt[y][0].get(x))
60.         totalnt += (totIGpnt[y][0].get(x))
61.     totalgain =
62. (pc)+(totpt.get(x)*totalt)+(totpnt.get(x)*totalnt)
63.     #print(x+" "+str(totalgain))
64.     nilaigain.update({x:totalgain})
65.     totalt = 0
66.     totalnt = 0
67.
68. for x in sortIG:
69.     for y in nilaigain:
70.         if(x==nilaigain.get(y)):
71.             selectedterm.update({y:nilaigain.get(y)})
72.
73. file_bio = open("IG.txt", "w")
74.
75. indeks = 0
76. jumlahkata = round(len(selectedterm)*(1/3))
77. for x in selectedterm:
78.     if(indeks<(jumlahkata-1)):
79.         print(x,selectedterm.get(x))
80.         file_bio.write(x+"|")
81.     elif(indeks==jumlahkata-1):
82.         print(x,selectedterm.get(x))
83.         file_bio.write(x)
84.     indeks+=1
```

**Source Code 5.2 Implementasi Proses Seleksi Fitur**

Penjelasan Source Code 5.2 Proses *Preprocessing*, baris ke:

1-13. Function untuk menghitung kemunculan sebuah kata pada kelas tertentu

15. Perulangan untuk menghitung peluang kemunculan kata pada dokumen setiap kelas dan ketidak munculan kata pada dokumen setiap kelas

17-22. Perulangan untuk menghitung peluang kemunculan kata dalam dokumen

23-33. Seleksi kondisi ketika nilai $P_Y(c_i|t)$ bernilai 0

34-37. Seleksi kondisi ketika nilai $P_Y(c_i|t)$ tidak bernilai 0

38-39. Seleksi kondisi ketika nilai $P_Y(c_i|\bar{t})$ bernilai 0

40-45. Seleksi kondisi ketika nilai $P_Y(c_i|\bar{t})$ tidak bernilai 0

48-53. Menyimpan hasil perhitungan peluang kemunculan kata

54-55. Mengosongkan dict

57. Perulangan untuk menghitung nilai information gain setiap term

58. Perulangan untuk menghitung nilai information gain setiap kelasnya

59-62. Menghitung total nilai information gain

64. Menyimpan nilai information gain

65-66. Mengosongkan nilai totalt dan totalnt

68. Perulangan untuk mengurutkan nilai information gain

69. Perulangan untuk mengurutkan seluruh term

70-71. Menyimpan term berdasarkan urutan nilai information gain tertinggi

75. Deklarasi variabel indeks

76. Deklarasi variabel jumlah kata untuk menentukan jumlah term yang digunakan untuk klasifikasi

77. Perulangan untuk menyimpan term *yang diseleksi*

*78-84. Seleksi kondisi untuk menyimpan term*

### 5.4.3 Implementasi Algoritme Pembobotan Kata

Proses pembobotan kata berguna untuk mengetahui bobot setiap kata, metode yang digunakan adalah TF-IDF-CF, dimana metode ini akan menghitung bobot setiap kata dengan cara menghitung jumlah kemunculan sebuah kata pada setiap dokumen, banyaknya dokumen yang terdapat sebuah kata, dan kemunculan kata pada setiap dokumen pada kelas tertentu.

```
1.  for x in dokumen:
2.      for y in x[0]:
3.          if str(dict.get(y))=='None':
4.              dict.update({y:1})
5.          else:
6.              dict[y]=dict.get(y)+1
7.      TF.append(dict)
8.      dict = {}
9.      data = []
10.
```

```
11.  b = []
12.  dokumenu = []
13.  for x in dokumen:
14.      for z in x[0]:
15.          if z not in b:
16.              b.append(z)
17.      dokumenu.append([b,x[1]])
18.      b = []
19.
20.  for x in range(len(dokumenu)):
21.      for k in dokumenu[x][0]:
22.          if (str(IDF.get(k))=='None'):
23.              IDF.update({k:1})
24.          else:
25.              IDF[k]=IDF.get(k)+1
26.
27.  kelas = []
28.  kls0 = []
29.  kls1 = []
30.  kls2 = []
31.  #Mengelompokkan Dokumen berdasarkan kelas
32.  for x in dokumen:
33.      #print(x)
34.      if(x[1]==0):
35.          kls0=kls0+x[0]
36.      elif(x[1]==1):
37.          kls1=kls1+x[0]
38.      elif(x[1]==2):
39.          kls2=kls2+x[0]
40.  kelas.append(kls0)
41.  kelas.append(kls1)
42.  kelas.append(kls2)
43.  dict = {}
44.
45.  #Menghiung CF
46.  for x in kelas:
47.      for k in x:
48.          if str(dict.get(k))=='None':
49.              dict.update({k:1})
50.          else:
51.              dict[k]=dict.get(k)+1
52.      CF.append(dict)
53.      dict={}
54.
55.  dwtf= {}
56.  dwtfcf={}
57.  dbIDF = client.GNB.IDF
58.  dbCF =  client.GNB.CF
59.
60.  #print(TFIDF)
61.  #Menghitung TF-IDF/TF-IDF-CF
62.  indeks = 0
63.  kategori = 0
64.  for x in TF:
65.      for y in x:
66.          if(indeks<50):
67.              kategori = 0
68.          elif(indeks>=50 and indeks<100):
69.              kategori=1
```

```
71.             else:
72.                 kategori = 2
73.             #print(x[y])
74.             #print(IDF.get(y))
75.             wtf =
76. (math.log10(x[y]+1))*(math.log10((float(len(dokumen)+1))/fl
77. oat(IDF.get(y))))
78.             wtfcf =
79. (math.log10(x[y]+1))*(math.log10((float(len(dokumen)+1))/fl
80. oat(IDF.get(y))))*(CF[kategori].get(y)/lenkelas)
81.
82. print(y,x[y],IDF.get(y),CF[kategori].get(y),wtf,wtfcf,kateg
83. ori)
84.             if(str(dbIDF.find_one({"term":y}))=="None"):
85.                 dbIDF.insert_one({"term" : y, "IDF" :
86. (math.log10((float(len(dokumen)+1))/float(IDF.get(y))))})
87.             dbCF.insert_one({"term" : y, "kelas" :
88. kategori,"CF" : (CF[kategori].get(y)/lenkelas)})
89.             dwtf.update({y:wtf})
90.             dwtfcf.update({y:wtfcf})
91.         TFIDF.append(dwtf)
92.         TF-IDF-CF.append(dwtfcf)
93.         dwtf={}
94.         dwtfcf={}
95.         indeks+=1
96.         print()
```

**Source Code 5.3 Implementasi Proses Pembobotan**

Penjelasan Source Code 5.3 Proses Pembobotan, baris ke:

1. Perulangan untuk menghitung nilai TF setiap kata pada dokumen
2. Perhitungan untuk menghitung kemunculan kata
3-4. Seleksi kondisi ketika belum terdapat sebuah kata pada dictionary dan untuk menambahkan nilai TF menjadi 1
5-6. Seleksi kondisi ketika sudah kata yang sama dalam dictionary dan memperbarui nilai TF
11-12. Inisialisasi variabel untuk menyimpan kata unik seluruh dokumen
13. Perulangan untuk menyimpan kata unik di setiao dokumen
14. Perulangan untuk memilih kata unik pada dokumen
15-16. Seleksi kondisi ketika belum terdapat kata pada b maka kata tersebut akan disimpan kedalam b
17. Menyimpan kata unik setiap dokumen
18. Mengosongkan variabel b
20. Perulangan untuk menghitung DF
21. Perulangan untuk menghitung DF setiap dokumen
22-23. Seleksi kondisi ketika belum terdapat sebuah kata pada dictionary dan untuk menambahkan nilai DF menjadi 1

24-25. Seleksi kondisi ketika sudah ada kata yang sama dalam dictionary dan memperbarui nilai DF

27-30 Deklarasi variabel yang akan digunakan untuk pengelompokkan dokumen berdasarkan kelas

32. Perulangan untuk mengelompokkan dokumen berdasarkan kelas

34-35. Seleksi kondisi ketika dokumen memiliki kelas 0 dan menambahkan dokumen tersebut kedalam kls0

36-37. Seleksi kondisi ketika dokumen memiliki kelas 1 dan menambahkan dokumen tersebut kedalam kls1

38-39. Seleksi kondisi ketika dokumen memiliki kelas 2 dan menambahkan dokumen tersebut kedalam kls2

40-42. Menyimpan hasil pengelompokkan dokumen kedalam 1 variabel

43. Mengosongkan dictionary

46. Perulangan untuk menghitung nilai CF

47. Perulangan untuk menghitung nilai CF setiap kelas

48-49. Seleksi kondisi ketika belum terdapat sebuah kata pada dictionary dan untuk menambahkan nilai CF menjadi 1

50-51. Seleksi kondisi ketika sudah ada kata yang sama dalam dictionary dan memperbarui nilai CF

52. Menyimpan nilai hasil perhitungan CF

53. Mengosongkan dictionary

55-56. Deklarasi variabel berupa dictionary untuk menyimpan hasil pembobotan

57-58. Deklarasi variabel untuk menyimpan database yang akan digunakan

62. Deklarasi variabel indeks dengan nilai 0

63. Deklarasi variabel kategori dengan nilai 0

64. Perulangan untuk menghitung term setiap dokumen

65. Perulangan menghitung bobot setiap term pada setiap dokumen

66-72. Seleksi kondisi untuk mengetahui kategori dokumen untuk disimpan pada database

75-83. Menghitung bobot nilai setiap term

84-88. Seleksi kondisi untuk menyimpan IDF dan CF

89-90. Menyimpan nilai hasil pembobotan kedalam dictionary

91-92. Menyimpan hasil pembobotan kedalam list

93-95. Mengosongkan nilai dictionary

95. Menambahkan 1 pada indeks disetiap perulangan

### 5.4.4 Implementasi Algoritme Naïve Bayes Classifier

Untuk melakukan klasifikasi pada data uji metode yang digunakan adalah *naïve bayes*, pada sistem yang dibuat bobot term menggunakan TFIDF/TF-IDF-CF sehingga memiliki nilai yang kontinyu, hal ini mengakibatkan metode *naïve bayes* yang digunakan adalah gaussian naïve bayes dimana *gaussian naïve bayes* bekerja menggunakan data bilangan kontinyu. Pada proses klasifikasi sendiri terdiri dari 2 proses yaitu *training* dan testing yang implementasinya dapat dilihat pada tabel dibawah.

#### 5.4.4.1 Implementasi Proses Training

Pada *gaussian naïve bayes* proses training berfungsi untuk menyimpan nilai prior setiap kategori, nilai mean, dan varians dari setiap term yang digunakan pada proses pengujian, nilai mean dan varians disimpan terlebih dahulu karena nilai mean dan varians dari sebuah term tidak akan berubah-ubah. Nilai mean dan varians akan disimpan dan akan digunakan pada proses pengujian untuk menghitung *likelihood.* Implementasi proses training dapat dilihat pada *Source Code* 5.4.

```
1.  dbprior = client.GNB.Prior
2.  dbprior.insert_one({"prior" : (lenkelas/len(dokumen))})
3.
4.  #Menghitung Mean dan Varians
5.  avg = 0
6.  indeks = 0
7.  kategori = 0
8.  total1= 0
9.  total2=0
10. dbMV =  client.GNB.MVTFIDF
11. for f in range(3):
12.     print("= - = - = - = - = - = - = - = - = - = - = - = - = -
13. =")
14.     for x in IDF:
15.         for y in range(indeks,indeks+50):
16.             if(str(TFIDF[y].get(x))=="None"):
17.                 avg += 0
18.                 total1+= (0)**2
19.                 total2+= (0)
20.             else:
21.                 avg += (TFIDF[y].get(x))
22.                 total1+= (TFIDF[y].get(x))**2
23.                 total2+= (TFIDF[y].get(x))
24.         dbMV.insert_one({"term" : x, "mean" :
25. (avg/lenkelas) , "varians" : ((lenkelas*total1)-
26. ((total2)**2))/(lenkelas*(lenkelas-1)), "kelas" : f})
27.         print(x,avg/lenkelas,((lenkelas*total1)-
28. ((total2)**2))/(lenkelas*(lenkelas-1)),f)
29.         total1=0
30.         total2=0
31.         avg=0
32.     indeks+=50
```

**Source Code 5.4 Implementasi Proses *Training***

Penjelasan Source Code 5.3 Proses *Training*, baris ke:

1. Deklarasi variabel untuk menyimpan database yang akan digunakan untuk menyimpan nilai prior
2. Menyimpan nilai prior untuk setiap kategori, hanya dilakukan sekali karena nilai prior pada setiap kategori

5-9. Deklarasi variabel yang akan digunakan untuk menghitung mean dan varians

10. Deklarasi variabel untuk menyimpan database yang akan digunakan untuk menghitun means dan varians
11. Perulangan untuk menghitung mean dan varians term setiap kelasnya

14. Perulangan untuk menghitung nilai mean dan varians setiap term pada dokumen

15. Perulangan untuk menghitung term pada kategori tertentu

16-19. Seleksi kondisi ketika tidak ada nilai bobot term pada dokumen

20-23. Seleksi kondisi ketika ada nilai bobot term pada dokumen90

24-26. Menyimpan hasil perhitungan mean dan varians sebuah term kedalam database

27-28. Menampilkan term dan hasil perhitungan varians dari term tersebut

29-31. Mengosongkan nilai variabel total1, total2, dan avg

32. Menambah 50 pada nilai indeks pada setiap perulangan

### 5.4.4.2 Implementasi Proses *Testing*

Langkah kedua pada proses *naïve bayes* yaitu proses *testing*, pada *gaussian naïve bayes*, dilakukan dengan menghitung bobot term terlebih dahulu, yang kemudian akan dihitung likelihood dari setiap term pada data uji, setelah itu menghitung nilai posterior untuk setiap kategori, kategori yang memiliki nilai posterior tertinggi akan menjadi kategori dari data yang diujikan

```
1.  def testing(uji,kelas):
2.
3.      posterior = []
4.      tweet = {}
5.      proses = {}
6.      total = 1
7.      for y in uji:
8.          if str(tweet.get(y))=='None':
9.              tweet.update({y:1})
10.         else:
```

```
            tweet[y]=tweet.get(y)+1

    for x in tweet:
        if(x in kata):
            proses.update({x:tweet.get(x)})

    #print(tweet)

    #print("= - = - = - = - = - = - = - = - = - = - =")
    #print(proses)

    #print(proses)
    total = 1
    for x in range(3):
     #   print("= - = - = - = - = - = - = - = - = - = - = - = - = - =")
        for y in proses:
            idf = IDF.find_one({"term":y})
            cf= CF.find_one({"term":y,"kelas":x})
            prior = dprior.find_one()
            #print(tf)
            if(str(cf)=="None"):
                total*=1
                #print(y,1)
            else:
                weight =(math.log10((proses.get(y))+1))*idf['IDF']*cf['CF']
                mevar = mv.find_one({"term":y,"kelas":x})
                mean = mevar['mean']
                var = mevar['varians']
                if(var==0):
                    var=0.00000000001

#print(y,(1/(math.sqrt(2*math.pi*var)))*((math.exp(1))**(-(((weight-mean)**2))/(2*var))))
                #print("lihat",weight,mean,var)
                #print((math.exp(1)**(-((tf['bobot']-mean)**2/(2*var)))))
                #print(var)

total*=(1/(math.sqrt(2*math.pi*var)))*((math.exp(1))**(-(((weight-mean)**2))/(2*var)))
                #print("var",((math.exp(1))**(-(((tf['bobot']-mean)**2))/(2*var))),(-(((tf['bobot']-mean)**2))/(2*var)))
        # print(prior['prior']*total,total)
         posterior.append(prior['prior']*total)
        # print(prior['prior']*total)
         total=1
    maks = 0
    kategori = 0
    for x in range(3):
        maks = max(maks,posterior[x])
        #print(maks)
    for x in range(3):
        if(posterior[x]==maks):
            kategori = x
```

**Source Code 5.5 Implementasi Proses *Testing***

Penjelasan *Source Code* 5.3 Proses *Training*, baris ke:

1. Deklarasi function untuk proses *testing*
3-6. Deklarasi variabel yang akan digunakan untuk pengujian
7. Perulangan untuk menyimpan TF term untuk diuji
10. Deklarasi variabel untuk menyimpan database yang akan digunakan untuk menghitun means dan varians
13-15. Menyeleksi term yang akan digunakan untuk pengujian
24. Perulangan untuk menghitung posterior setiap kategori
27. Perulangan untuk menghitung likelihood setiap term pada data uji yang sudah diseleksi
28-30. Deklarasi variabel untuk mengambil nilai IDF, CF, dan prior dari database
32-33. Seleksi kondisi jika term tidak memiliki nilai CF
35-40. Seleksi kondisi jika term memiliki nilai CF
51-52. Menghitung nilai total likelihood
57. Menghitung dan menyimpan nilai posterior setiap kategori
58. Mengubah nilai variabel total menjadi 1
60-61. Mengubah nilai variabel maks dan kategori menjadi 0
62. Perulangan untuk mencari nilai posterior tertinggi dari setiap kategori
63. Menggunakan fungsi max untuk mencari nilai tertinggi
65. Perulangan untuk membandingkan nilai tertinggi dengan kategori
66. Seleksi kondisi ketika nilai posterior tertinggi sama dengan nilai posterior pada kategori tertentu
67. Menyimpan nilai kategori

# BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini menjelaskan tentang pengujian apa saja yang dilakukan dan analisis terhadap hasil implementasi dan pengaruh metode yang digunakan terhadap akurasi dari mesin klasifikasi.

## 6.1 Pengujian pengaruh Information Gain

### 6.1.1 Skenario Pengujian Pengaruh Information Gain

Pengujian ini dilakukan untuk mengetahui pengaruh dari seleksi fitur IG terhadap akurasi klasifikasi, pengujian ini pertama dilakukan tanpa menggunakan seleksi fitur kemudian akan menggunakan seleksi fitur IG dengan pemilihan jumlah term yang akan digunakan pada proses klasifikasi yaitu 10%, 15%, 20%, 33%, 66%, dan 80%, IG sendiri bekerja dengan menyeleksi term dari data latih dengan memilih term dengan nilai IG tertinggi sejumlah dengan yang sudah ditentukan. Term yang sudah diseleksi tadi akan dibandingkan dengan term pada data uji ketika melakukan klasifikasi term yang digunakan pada pada proses klasifikasi hanyalah term data uji yang terdapat dari hasil seleksi IG. Hasil pengujian pengaruh IG dapat dilihat pada Tabel 6.1. hingga 6.7.

**Tabel 6.1 Pengujian menggunakan jumlah term sebesar 10%**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 12 | 8 | 13 | 7 | 14 | 5 |

**Tabel 6.2 Pengujian menggunakan jumlah term sebesar 15%**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 11 | 9 | 15 | 5 | 14 | 6 |

**Tabel 6.3 Pengujian menggunakan jumlah term sebesar 20%**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 12 | 8 | 14 | 6 | 15 | 5 |

**Tabel 6.4 Pengujian menggunakan jumlah term sebesar 33%**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 11 | 9 | 12 | 8 | 14 | 6 |

**Tabel 6.5 Pengujian menggunakan jumlah term sebesar 66%**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 11 | 9 | 7 | 13 | 14 | 6 |

**Tabel 6.6 Pengujian menggunakan jumlah term sebesar 80%**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 11 | 9 | 9 | 11 | 11 | 9 |

**Tabel 6.7 Pengujian tanpa menggunakan *Information Gain***

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 10 | 10 | 12 | 8 | 6 | 14 |

## 6.1.2 Analisis Pengujian Pengaruh Information Gain

Merujuk pada Tabel 6.1, 6.2, 6.3, 6.4, 6.5, 6.6, dan 6.7 dapat diilhat saat tanpa menggunakan IG hasil klasifikasi hanya mampu mendapatkan hasil akurasi sebesar sekitar 46%, hal ini terjadi karena penelitian ini mengklasifikasikan kategori kontrakkan, kost putra, dan kost putri, pada penelitian ini pada setiap kategori akan memiliki term yang lebih merepresentasikan setiap kategorinya, tapi pada kenyataannya setelah dilakukan penelitian hasil akurasi hanya mencapai 46 % hal ini dikarenakan memang pada seluruh kategori tersebut memiliki term atau kata-kata yang bisa dibilang memiliki kemunculan pada setiap kelasnya

sehingga tidak menutup kemungkinan hasil dari klasifikasi mendapatkan hasil yang kurang baik, karena term didalam setiap kategorinya yang cukup mirip, meski memang ada beberapa term yang lebih merepresentasikan kelasnya, pada Tabel 6.5 diberikan contoh beberapa term yang berada pada beberapa kelas.

**Tabel 6.8 Contoh Term yang muncul pada beberapa kelas**

| Term | Term Frekuensi | | |
|---|---|---|---|
| | Kontrakkan | Kost Putra | Kost Putri |
| Kamar | 26 | 39 | 28 |
| Tidur | 4 | 1 | 1 |
| Mandi | 10 | 14 | 12 |
| Putra | 3 | 49 | 0 |
| Info | 17 | 2 | 5 |

Pada penggunaan term sebesar 20% hasil klasifikasi meningkat cukup signifikan menghasilkan akurasi tertinggi pada pengujian yang pertama ini yaitu sekitar 68,33%, hal ini terjadi karena memang pada seluruh kategori yang ada tidak banyak memiliki term yang lebih merepresentasikan setiap kategorinya, karena itu dengan pemilihan term yang bisa dibilang cukup sedikit hanya dengan 20% memiliki hasil akurasi yang meningkat, terutama pada data uji dikategori Kost Putri, kategori ini meningkat sangat tinggi, hal ini dikarenakan pada pengujian tanpa IG kemungkinan banyak term yang muncul dikategori lainnya, terutama pada kelas kost putra sehingga mengakibatkan hasil klasifikasi yang kurang baik ketika tanpa menggunakan IG, ketika menggunakan 20% saja kemungkinan term-term yang kurang merepresentasikan kategori masing-masing akan hilang yang mengakibatkan peningkatan yang cukup signifikan.

Namun ketika dilakukan pengujian dengan penggunaan threshold dibawah 20% yaitu, 15% dan 10% terjadi penurunan hasil akurasi, hal ini dikarenakan ketika penggunaan term sangat sedikit akan mengakibatkan berkurangnya juga term-term yang merepresentasikan kelas tertentu, dapat dilihat pada tabel 6.3 dan 6.2, terjadi penurunan akurasi ketika dilihat pada kategori kontrakkan dan kategori kost putri, karena semakin sedikitnya term yang merepresentasikan kelas-kelas tersebut, meski terdapat peningkatan akurasi pada kategori kost putra, namun lebih banyak terjadi penurunan ketika penggunaan term dibawah 20%.

### 6.1.3 Skenario Pengujian Pengaruh TF-IDF-CF

Pengujian ini dilakukan untuk mengetahui pengaruh dari parameter CF pada pembobotan TF-IDF, cara kerjanya berkebalikan dengan IDF dimana IDF akan memiliki nilai yang lebih kecil dengan mempertimbangkan banyaknya dokumen yang terdapat suatu term, sedangkan CF akan memiliki nilai lebih besar jika banyak dokumen yang terdapat suatu term pada suatu kategori. Pada skenario pengujian ini tetap menggunakan variasi IG seperti pengujian sebelumnya, namun menggunakan pembobotan TF-IDF-CF. Hasil pengujian menggunakan TF-IDF-CF dapat dilihat pada Tabel 6.9, 6.10, 6.11, 6.12, 6.13, 6.14, 6.15.

**Tabel 6.9 Pengujian menggunakan jumlah term sebesar 10% dan TF-IDF-CF**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 14 | 7 | 11 | 9 | 15 | 5 |

**Tabel 6.10 Pengujian menggunakan jumlah term sebesar 15% dan TF-IDF-CF**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 13 | 7 | 14 | 6 | 15 | 5 |

**Tabel 6.11 Pengujian menggunakan jumlah term sebesar 20% dan TF-IDF-CF**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 13 | 7 | 13 | 7 | 16 | 4 |

**Tabel 6.12 Pengujian menggunakan jumlah term sebesar 33% dan TF-IDF-CF**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 12 | 8 | 13 | 7 | 14 | 6 |

**Tabel 6.13 Pengujian menggunakan jumlah term sebesar 66% dan TF-IDF-CF**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 11 | 9 | 8 | 12 | 14 | 6 |

**Tabel 6.14 Pengujian menggunakan jumlah term sebesar 80% dan TF-IDF-CF**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 11 | 9 | 9 | 11 | 12 | 8 |

**Tabel 6.15 Pengujian tanpa menggunakan *Information Gain* dan TF-IDF-CF**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 12 | 8 | 12 | 8 | 6 | 14 |

### 6.1.4 Analisis Pengujian Pengaruh TF-IDF-CF

Terbukti dengan adanya CF dapat dilihat dari hasil pengujian pada Tabel 6.5 hingga 6.8 pada setiap percobaan dengan IG yang berbeda selalu ada peningkatan dari yang hanya menggunakan TFIDF menjadi TF-IDF-CF, hal ini dikarenakan TF-IDF-CF akan mengakibatkan term yang relevan pada suatu kelas akan memiliki nilai yang lebih tinggi, dibandingkan dengan TFIDF yang hanya mempertimbangkan kemunculan term pada sebuah dokumen dan banyaknya dokumen yang terdapat sebuah term, tanpa mempertimbangkan pada kategori mana term tersebut muncul, dengan begitu peningkatan terjadi karena kesalahan klasifikasi ketika menggunakan TFIDF hanya memiliki selisih yang tidak terlalu besar, dapat dilihat pada Tabel 6.10 dengan menggunakan term sebesar 15% dimana pada pengujian sebelumnya memiliki akurasi dibawah penggunaan term sebesar 20% ketika menggunaan TF-IDF-CF akurasinya mengalami peningkatan hingga mampu menyamai akurasi terbaik ketika menggunakan term sebesar 20% yaitu didapatkan akurasi sebesar 70%. Hal ini berarti dengan menggunakan TF-IDF-CF mampu meningkatkan akurasi dari hasil klasifikasi, hal ini dikarenakan TFIDF

mampu meningkatkan bobot term yang relevan pada sebuah kategori dan akan menurunkan bobot term yang tidak atau kurang relevan pada sebuah kategori.

**Tabel 6.16 Perbandingan akurasi menggunakan TF-IDF dan TF-IDF-CF**

| Jumlah term pada seleksi fitur | Akurasi | |
|---|---|---|
| | TF-IDF | TF-IDF-CF |
| 10% | 65% | 66,66% |
| 15% | 66,66% | 70% |
| 20% | 68,33%% | 70% |
| 33% | 61,66% | 65% |
| 66% | 53,33% | 55% |
| 80% | 51,66% | 53,33% |
| Tanpa seleksi fitur | 46,66% | 50% |

## 6.1.5 Skenario Pengujian Tanpa Fitur Angka

Pengujian ini dilakukan untuk mengetahui pengaruh fitur berupa angka, pengujian ini dilakukan untuk mengetahui pengaruh fitur angka terhadap hasil pengklasifikasian, pengujian ini dilakukan dengan menggunakan seleksi fitur dengan penggunaan term optimal dan pemobotan terbaik, sehingga dipilih penggunaan term sebesar 20% dan pembobotan TF-IDF-CF. Untuk hasil pengklasifikasian sendiri dapat dilihat pada tabel 6.17.

**Tabel 6.17 Pengujian tanpa menggunakan *Information Gain* dengan TF-IDF-CF**

| Jumlah data uji | Kategori | | | | | |
|---|---|---|---|---|---|---|
| | Kontrakan | | Kost Putra | | Kost Putri | |
| | Benar | Salah | Benar | Salah | Benar | Salah |
| 60 | 14 | 6 | 14 | 6 | 15 | 5 |

## 6.1.6 Analisis Pengujian Tanpa Fitur Angka

Terjadi peningkatan akurasi ketika proses pengklasifikasian dilakukan tanpa menggunakan fitur angka, hal ini terjadi dikarenakan fitur angka pada penelitian ini tidak diperlukan, karena fitur angka memiliki kemunculan di beberapa kelas, karena hal tersebut meski angka-angka tersebut memiliki banyak kemunculan, namun fitur angka ini bisa dibilang tidak merepresentasikan kelas manapun sehingga ketika dilakukan pengujian tanpa menggunakan fitur angka ini terjadi peningkatan akurasi dari 70% menjadi 71,66%.

# BAB 7 PENUTUP

## 7.1 Kesimpulan

Berdasarkan pengujian yang telah dilakukan sebelumnya dan analisis yang telah dilakukan dapat ditarik kesimpulan dibawah ini.

1. Akurasi yang didapatkan menggunakan metode *naïve bayes*, khususnya *gaussian naïve bayes*, bisa dibilang masih kurang baik karena berkaca pada beberapa penelitian sebelumnya pada penelitian ini hanya mampu mencapai akurasi terbaik sebesar 71,66%. Hal ini dikarenakan pada seluruh kategori yang digunakan memiliki banyak term yang sama didalam kategorinya sehingga memungkinkan kesalahan dalam melakukan klasifikasi.
2. Penggunaan seleksi fitur IG dapat meningkatkan akurasi dari mesin klasifikasi, hal ini dikarenakan IG bekerja dengan menyeleksi term yang ada dengan menghasilkan nilai IG yang lebih tinggi ketika sebuah term banyak muncul hanya pada 1 kategori dibandingkan term yang banyak muncul di beberapa kategori, pada cukup banyak term yang muncul pada beberapa kategori sehingga dengan adanya IG mampu memilih term yang relevan dengan kategorinya sendiri, pada kasus ini karena banyak term yang sama muncul pada beberapa kategori mengakibatkan dengan presentasi penggunaan term semakin kecil didapatkan akurasi yang semakin tinggi, pada penggunaan 20% term mampu menghasilkan akurasi sebesar 68,3 %, sedangkan pada penggunaan term diatas 20% akurasi hanya mencapai angka 61,66%, 53% dan 51.6%. Ketika term yang digunakan dibawah 20% terdapat penurunan akurasi yang dikarenakan semakin sedikitnya term yang relevan pada setiap kelasnya akurasi yang didapatkan ketika menggunakan term sebesar 15% dan 10% adalah 66,66% dan 65%. Sedangkan pengaruh TF-IDF-CF sangat terlihat pada peningkatan akurasi pada setiap percobaan menggunakan variasi IG yang sama TF-IDF-CF mampu meningkatkan setiap akurasi pada pengujian dimana pada pengujian dengan penggunaan term 33% akurasi menjadi 65%, pada penggunaan 66% term akurasi menjadi 61.6%, dan pada pengujian dengan penggunaan 80% term akurasi menjadi 63%

## 7.2 Saran

1. Pada IG masih belum mampu membedakan jumlah penggunaan term perkategorinya sehingga tidak menutup kemungkinaan term yang diseleksi mendapatkan presentasi yang lebih besar disbanding kategori lain, oleh karena itu pada penelitian selanjutnya perlu metode yang mampu menyeimbangkan penggunaan term pada setiap kategorinya.
2. Pada penelitian selanjutnya mungkin dapat dilakukan pengujian menggunakan metode klasifikasi lain, atau bahkan menggunakan metode naïve bayes lain seperti multinomial atau bernoulli.

3. Diperlukan pengujian menggunakan metode klasifikasi lain untuk mengetahui nilai akurasi menggunakan TF-IDF-CF.

4. Penelitian ini mendapatkan hasil klasifikasi terbaik ketika menggunakan term sebesar 20% yang memperlihatkan bahwa term yang merepresentasikan tidak cukup banyak, sehingga pada penelitian selanjutnya bisa dilakukan *generate role* otomatis salah satunya menggunakan *fuzzy inference* untuk memilih term-term yang digunakan sebagai fitur yang paling relevan.

5. Pada penelitian ini dapat dilihat karena term yang kurang baku dapat mengakibatkan pengklasifikasian yang kurang baik, sehingga pada penelitian selanjutnya dapat dilakukan normalisasi kata atau mengubah kata yang tidak baku menjadi kata baku.

# DAFTAR REFERENSI


Abidin, A, 2017. Lima Tahun, Penduduk Kota Malang Bertambah 50.116 Orang, Tribun Jatim, http://jatim.tribunnews.com/2017/02/14/lima-tahun-penduduk-kota-malang-bertambah-50116-orang. (diakses pada 9 september 2018)

Buzic, D., Dobsa, J.,2018. *Lyrics Classification using Naive Bayes*, *International Convention on Information and Communication Technology, Electronics and Microelectronics*, 41, pp.1011-1015

Capdevila M, Flórez Oscar W. M., 2009, *A Communication Perspective on Automatic Text Categorization*, I*EEE Transactions on Knowledge and Data Engineering*, 21, pp.1027-1041

Chiang, O., 2011. *Twitter hits nearly 200M accounts, 110M tweets per day, focuses on global expansion. Forbes.* http://blogs.forbes.com/oliverchiang/2011/01/19/twitter-hits-nearly-200m-users-110m-tweets-per-day-focuses-on-global-expansion/.(diakses 2 September 2018)

Chormunge, S., Jena, S., 2016. *Efficient Feature Subset Selection Algorithm for High Dimensional Data*. *International Journal of Electrical and Computer Engineering*, 6, pp. 1880-1888

Fatahillah, N., Suryati, P., Haryawan, C., 2017, *Implementation Of Naive Bayes Classifier Algorithm On Social Media (Twitter) To The Teaching Of Indonesian Hate Speech*, *International Conference on Sustainable Information Engineering and Technology (SIET)*, pp. 128-131

Feldman R., Sanger J., 2007. *The Text Mining Handbook Advanced Approaches in Analyzing Unstructured Data.* New York: Cambridge University Press

Firmahsyah, Gantini, T., 2016. Penerapan Metode *Content-Based Filtering* Pada Sistem Rekomendasi Kegiatan Ekstrakulikuler (Studi Kasus di Sekolah ABC). Jurnal Teknik Informatika dan Sistem Informasi, 2, pp. 414-427

Forman G., 2003. *An Extensive Empirical Study of Feature Selection Metrics for Text Classification*, *Journal of Machine Learning Research*, 3, pp. 1289-1305

Hamzah, A., 2012. Klasifikasi Teks dengan *Naïve Bayes Classifier* untuk Pengelompokkan Teks Berita dan *Abstract* Akademis. Prosiding Seminar Nasional Aplikasi Sains & Teknologi, 3, pp. 269-277

Lewis, David D. 1998. Naive (Bayes) at forty: The independence assumption in information retrieval. *European Conference on Machine Learning,* pp. 4-15

Liu, M., Yang, J., 2012) *An Improvement of TFIDF weighting in text categorization*. *International Proceedings of Computer Science and Information Technology,* 47, pp. 44-47

Putra, I., Sudarma, M., Kumara, I., 2016, Klasifikasi Teks Bahasa Bali Dengan *Metode Supervised Learning Naive Bayes Classifier*, Teknologi Elektro, 14, pp. 81-86

Triawati, Chandra. 2009. Metode Pembobotan *Statistical Concept Based* untuk Klastering dan Kategorisasi Dokumen Berbahasa Indonesia. Bandung: Institut Teknologi Telkom.

Wulandini F., Nugroho Anto S., 2009. *Text Classification Using Support Vector Machine for Webmining Based Spatio Temporal Analysis of the Spread of Tropical Diseases*. *International Conference on Rural and Communication Technology*, 1, pp 189-192