**ANALISIS SENTIMEN MENGENAI KULIAH DARING DI TWITTER MENGGUNAKAN METODE *NAÏVE BAYES* DAN PEMBENTUKAN *STOPWORD* DENGAN *TERM BASED RANDOM SAMPLING***

PROPOSAL SKRIPSI

Disusun oleh:

Raditya Rinandyaswara

NIM: 175150200111047



PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

MALANG

2020

DAFTAR ISI

[DAFTAR ISI ii](#_Toc54440417)

[DAFTAR TABEL iv](#_Toc54440418)

[DAFTAR GAMBAR v](#_Toc54440419)

[DAFTAR LAMPIRAN vi](#_Toc54440420)

[BAB 1 PENDAHULUAN 1](#_Toc54440421)

[1.1 Latar Belakang 1](#_Toc54440422)

[1.2 Rumusan Masalah 2](#_Toc54440423)

[1.3 Tujuan 2](#_Toc54440424)

[1.4 Manfaat 2](#_Toc54440425)

[1.5 Batasan Masalah 3](#_Toc54440426)

[1.6 Sistematika Pembahasan 3](#_Toc54440427)

[BAB 2 LANDASAN KEPUSTAKAAN 5](#_Toc54440428)

[2.1 Kajian Pustaka 5](#_Toc54440429)

[2.1.1 *New Normal* 6](#_Toc54440430)

[2.1.2 Kuliah Daring 6](#_Toc54440431)

[2.1.3 Twitter 7](#_Toc54440432)

[2.2 Teks *Pre-processing* 7](#_Toc54440433)

[2.2.1 Case Folding 7](#_Toc54440434)

[2.2.2 Cleaning 7](#_Toc54440435)

[2.2.3 Tokenisasi 7](#_Toc54440436)

[2.2.4 *Stopword* Removal 7](#_Toc54440437)

[2.2.5 Stemming 7](#_Toc54440438)

[2.3 *Term Based Random Sampling* 8](#_Toc54440439)

[2.4 Pembobotan Kata (TF-IDF) 9](#_Toc54440440)

[2.5 Algoritme *Naïve Bayes* 9](#_Toc54440441)

[2.6 *Confusion Matrix* 10](#_Toc54440442)

[2.7 *K-Fold Cross Validation* 11](#_Toc54440443)

[BAB 3 METODOLOGI 13](#_Toc54440444)

[3.1 Tipe Penelitian 13](#_Toc54440445)

[3.2 Strategi Penelitian 13](#_Toc54440446)

[3.3 Subjek Penelitian 13](#_Toc54440447)

[3.4 Peralatan Pendukung 13](#_Toc54440448)

[3.5 Lokasi Penelitian 14](#_Toc54440449)

[3.6 Teknik Pengumpulan Data 14](#_Toc54440450)

[3.7 Data Penelitian 14](#_Toc54440451)

[3.8 Teknik Analisis Data 14](#_Toc54440452)

[3.9 Implementasi Algoritme 14](#_Toc54440453)

[BAB 4 Perancangan 15](#_Toc54440454)

[4.1 Diagram Alir Sistem 15](#_Toc54440455)

[4.1.1 Diagram Alir *Preprocessing* tanpa Filtering 16](#_Toc54440456)

[4.1.2 Diagram Alir *Term Based Random Sampling* 17](#_Toc54440457)

[4.1.2.1 Diagram Alir Kullback Leibler Divergence 21](#_Toc54440458)

[4.1.2.2 Diagram Alir Normalisasi MinMax Term Weighting 22](#_Toc54440459)

[4.1.3 Diagram Alir *Preprocessing* 23](#_Toc54440460)

[4.1.4 Diagram Alir *Term Weighting* 24](#_Toc54440461)

[4.1.4.1 Diagram Alir Raw Term Weighting 25](#_Toc54440462)

[4.1.4.2 Diagram Alir Log Term Weighting 26](#_Toc54440463)

[4.1.4.3 Diagram Alir Inverse Document Frequency 27](#_Toc54440464)

[4.1.4.4 Diagram Alir Term Frequency - Inverse Document Frequency 28](#_Toc54440465)

[4.1.5 Diagram Alir *Naïve Bayes Training* 29](#_Toc54440466)

[4.1.5.1 Diagram Alir Hitung Likelihood term tiap kelas 31](#_Toc54440467)

[4.1.5.2 Diagram Alir Hitung Prior tiap kelas 32](#_Toc54440468)

[4.1.6 Diagram Alir *Naïve Bayes Testing* 32](#_Toc54440469)

[DAFTAR REFERENSI 33](#_Toc54440470)

DAFTAR TABEL

[Tabel 2.1 Confusion Matrix 10](#_Toc54432534)

[Tabel 3.1 Spesifikasi Hardware 13](#_Toc54432535)

[Tabel 3.2 Spesifikasi Software 13](#_Toc54432536)

DAFTAR GAMBAR

[Gambar 2.1 Ilustrasi *K-Fold* *Cross Validation* 11](#_Toc54440471)

[Gambar 4.1 Diagram Alir Sistem, (a) Tahap Pembuatan Stopword, (b) Tahap Pelatihan, (c) Tahap Pengujian 15](#_Toc54440472)

[Gambar 4.2 Diagram Alir Preprocessing tanpa Filtering 17](#_Toc54440473)

[Gambar 4.3 Diagram Alir *Term Based Random Sampling* 21](#_Toc54440474)

[Gambar 4.4 Diagram Alir Kullback Leibler Divergence 22](#_Toc54440475)

[Gambar 4.5 Diagram Alir Normalisasi MinMax Term Weighting 23](#_Toc54440476)

[Gambar 4.6 Diagram Alir Preprocessing 24](#_Toc54440477)

[Gambar 4.7 Diagram Alir Term Weighting 25](#_Toc54440478)

[Gambar 4.8 Diagram Alir *Raw Term Wighting* 26](#_Toc54440479)

[Gambar 4.9 Diagram Alir *Log Term Wighting* 27](#_Toc54440480)

[Gambar 4.10 Diagram Alir *Inverse Document Frequency* 28](#_Toc54440481)

[Gambar 4.11 Diagram Alir *Term Frequency - Inverse Document Frequency* 29](#_Toc54440482)

[Gambar 4.12 Diagram Alir *Naive Bayes Training* 31](#_Toc54440483)

[Gambar 4.13 Diagram Alir Hitung Likelihood term tiap kelas 31](#_Toc54440484)

[Gambar 4.14 Diagram Alir Hitung Prior tiap kelas 32](#_Toc54440485)

DAFTAR LAMPIRAN

# PENDAHULUAN

Bab ini terdiri dari hal yang melatarbelakangi dari penelitian ini dilaksanakan, rumusan masalah yang diperoleh dari latar belakang hingga tujuan dan manfaat dari penelitian ini serta batasan yang dijabarkan sesuai dengan cakupan dan kemampuan penulis , maupun sistematika yang menuliskan secara rangkum isi dari tiap bab.

## Latar Belakang

Pada saat ini kemajuan teknologi di dunia maupun di Indonesia mengalami perkembangan yang cukup pesat. Kemajuan teknologi ini salah satunya adalah ditandai dengan mudahnya proses pertukaran Informasi antara satu pengguna dan pengguna lainnya. Kemudahan proses pertukaran informasi ini ditandai dengan maraknya bermuncul sosial media di Internet. Twitter merupakan salah satu contoh dari sosial media yang banyak digunakan oleh masyarakat Indonesia yang digunakan sebagai sarana pertukaran informasi di dunia digital. Dalam pengunaannya, Twitter memberi istilah kepada pertukaran informasi tersebut dengan nama *Tweets,* yang mana *Tweets* adalah suatu teks atau kata yang dibatasi panjang nya hingga 280 karakter yang nanti akan di-*posting* dalam platform Twitter tersebut.

Pada akhir tahun 2019 lalu, dunia dikejutkan dengan adanya wabah yang diakibatkan oleh virus corona yang berasal dari kota Wuhan, China. Penyakit yang disebut sebagai (COVID-19) ini adalah penyakit yang menyerang sistem pernapasan virus manusia. Menurut data pemerintah China, penduduk Hubei menjadi kasus pertama Covid-19 pada 17 November 2019 (Arnani, 2020). Setelah kasus pertama Covid-19 di dunia itu terjadi peningkatan pasien tiap bulannya. Hingga saat ini Indonesia sudah melewati angka 190 ribu kasus Covid-19 yang sudah terkonfirmasi yang terhitung dari sejak pasien pertama (Ramadhan, et al., 2020). Dengan adanya pandemi Covid-19 ini, pemerintah Indonesia mengadakan sistem *New Normal* dengan tujuan untuk mempercepat penanganan Covid-19 (Putsanra, 2020). Dalam penerapannya banyak perilaku yang biasa dilakukan secara luring berubah menjadi daring. Salah satu contohnya adalah perkuliahan. Tentunya kuliah daring ini menyebabkan banyak pro dan kontra dan salah satu sarana masyarakat menuliskan opininya mengenai kuliah daring tersebut yaitu melalui Twitter. Namun twitter sendiri hanya berfungsi sebagai sosial media sehingga tidak menyediakan analisis sentimen dari *tweet* pengguna dan bagaimana masyarakat Indonesia menanggapi protokol baru tersebut.

Analisis Sentimen atau *Opinion Mining* adalah salah satu bidang studi yang menganalisis pendapat, sentimen, evaluasi, penilaian, sikap, dan emosi orang terhadap entitas seperti produk, layanan, organisasi, isu, peristiwa, topik, dan atributnya (Liu, 2012). Dengan proses analisis sentimen ini kita dapat mengetahui bagaimana pendapat orang apakah cenderung positif, negatif atau pun netral.

Pada penelitian ini, peneliti akan membangun sebuah sistem analisis sentimen terhadap kuliah daring yang dituliskan masyarakat di Twitter menggunakan metode klasifikasi *Naïve Bayes*. Dalam penelitian ini klasifikasi akan dibuat menjadi 3 kelas yaitu opini netral, positif dan negatif sesuai dari saran penelitian analisis sentimen sebelumnya (Sa'rony, et al., 2019). Dalam proses analisis sentimen diperlukan *preprocessing* data agar data dapat dihitung dan diolah*.* Salah satu tahap yang penting dalam *preprocessing* adalah *Stopword Removal*, yaitu penghapusan kata yang tidak memiliki nilai keunikan dalam suatu dokumen. Biasanya dalam proses penghapusan *stopword* ini memanfaatkan *digital library* namun ada *stopword* yang kurang tepat sehingga hal itu dapat membuat kualitas proses *stopword* removal menjadi berkurang (Dila Purnama Sari, et al., 2020). Oleh karena itu perlu adanya suatu mekanisme sistem daftar *stopword* dinamik yang dapat menghasilkan daftar *stopword* yang sesuai dengan yang diperlukan sesuai saran dari penelitian sebelumnya (Rahutomo & Ririd, 2018). Oleh karena itu pada penelitian ini, peneliti akan melakukan pembuatan daftar *stopword* menggunakan metode *Term Based Random Sampling*.

## Rumusan Masalah

Dari latar belakang yang telah diuraikan sebelumnya, berikut adalah rumusan masalah untuk penelitian ini:

1. Bagaimana pengaruh parameter X, Y dan L pada *Term Based Random Sampling* terhadap hasil pembentukan *stopword*?
2. Bagaimana hasil evaluasi dari pembentukan *stopword Term Based Random Sampling* pada analisis sentimen dengan *Naïve Bayes*?
3. Bagaimana perbandingan hasil evaluasi dari pembentukan *stopword* dengan *Term Based Random Sampling* dengan *stopword* Tala pada analisis sentimen dengan Naïve Bayes?

## Tujuan

Tujuan penelitian ini adalah:

1. Mengetahui pengaruh parameter X, Y dan L pada *Term Based Random Sampling* terhadap hasil pembentukan *stopword*.
2. Mengetahui hasil evaluasi dari pembentukan *stopword Term Based Random Sampling* pada analisis sentimen dengan Naïve Bayes.
3. Mengetahui perbandingan hasil evaluasi dari pembentukan *stopword* dengan *Term Based Random Sampling* dengan *stopword* Tala pada analisis sentimen dengan Naïve Bayes.

## Manfaat

Manfaat penelitian ini adalah:

1. Dapat mengetahui bagaimana pengaruh parameter X, Y dan L pada *Term Based Random Sampling* terhadap hasil pembentukan *stopword*
2. Dapat mengetahui dari pembentukan stopword Term Based Random Sampling pada analisis sentimen dengan Naïve Bayes.
3. Dapat mengetahui perbandingan hasil evaluasi dari pembentukan stopword dengan Term Based Random Sampling dengan stopword Tala pada analisis sentimen dengan Naïve Bayes.

## Batasan Masalah

Batasan masalah penelitian ini adalah:

1. Hanya menggunakan opini pengguna Twitter mengenai Kuliah Daring.
2. Algoritma yang digunakan hanya *Naïve Bayes Classifier* tidak membandingkan dengan algoritma lain.
3. Hasil klasifikasi sentimen hanya dibagi menjadi tiga kelas yaitu positif, netral, dan negatif.
4. *Tweet* yang merupakan data hanya *tweet* yang berbahasa Indonesia.
5. Jumlah data yang digunakan sebanyak 300 data.

## Sistematika Pembahasan

Berikut sistematika pembahasan dalam penelitian ini:

BAB I PENDAHULUAN

Bab ini menjelaskan mengenai latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika pembahasan sesuai dengan aturan dalam peneliltian

BAB II LANDASAN KEPUSTAKAAN

Landasan Kepustakaan menjelaskan penelitian-penelitian sebelumnya yang serupa dengan penelitian dalam proposal ini, serta dasar-dasar teori yang akan di implementasikan dalam penelitian ini seperti *preprocessing, term weighting, Naïve Bayes, Term Based Random Sampling,* serta tabel *confusion matrix* sehingga dapat mendukung penelitian.

BAB III METODOLOGI

Pada bab ini dijelaskan tentang bagaimana menerapkan penelitian seperti untuk mengimplementasikan *Naïve Bayes* dengan pembuatan daftar *Stopword* untuk analisis sentimen pengguna Twitter terhadap layanan Kuliah Daring.

BAB IV PERANCANGAN

Bab ini menjelaskan bagaimana proses perancangan dalam sistem yang akan dibangun.

BAB V IMPLEMENTASI

Pada bab ini menjelaskan bagimana implementasi sistem yang sudah dirancang di bab sebelumnya.

BAB VI PENGUJIAN DAN ANALISIS

Pada bab ini dilakukan pengujian terhadap sistem yang sudah dibangun dan menganalisa hasil yang didapatkan untuk menemukan kesimpulan dari hasil pengujian.

BAB VII PENUTUP

Pada bab terakhir ini menjelaskan tentang bagaimana kesimpulan dari penelitian ini dan saran untuk penelitian berikutnya.

# LANDASAN KEPUSTAKAAN

Pada bab ini akan dijelaskan bagaimana penelitian-penelitian sebelumnya yang serupa dengan penelitian yang sedang diajukan, serta dasar-dasar teori yang akan diimplementasikan dalam penelitian ini seperti *preprocessing, term weighting, Naïve Bayes, Term Based Random Sampling,* serta tabel *confusion matrix* yang dapat mendukung penelitian.

## Kajian Pustaka

Pada bagian ini akan dibahasnya mengeni penelitian atau kajian pustaka yang digunakan dalam penelitian ini berasal dari beberapa penelitian yang sudah dilakukan sebelumnya dan memiliki keterkaitan dengan judul skripsi Analisis Sentimen mengenai Kuliah Daring di Twitter menggunakan metode *Naïve Bayes* dan Pembentukan *Stopword* dengan *Term Based Random Sampling*.

Contoh salah satu penelitian terkait judul skripsi yang telah disebutkan adalah penelitian yang dilakukan oleh (Septian, et al., 2019) yaitu mengenai analisis sentimen pengguna Twitter terhadap polemik persepakbolaan Indonesia menggunakan pembobotan TF-IDF dan metode klasifikasi *K-Nearest Neighbor*. Penelitian ini menggunakan kamus kata tidak baku yang dibuat oleh peneliti secara manual yang nantinya akan digunakan sebagai normalisasi kata. Hasil pengujian yang didapatkan dari pengujian silang sebanyak 10 kali dan mendapatkan hasil akurasi optimal pada nilai k-23 sejumlah 79.99%.

Selain *K-Nearest Neighbor*, salah satu metode klasifikasi umum yang digunakan adalah *Naïve Bayes*. Pada penelitian sebelumnya oleh (Devita, et al., 2018) Kinerja metode *Naïve Bayes* dibandingkan dengan *K-Nearest Neighbor* untuk Klasifikasi Teks Artikel berbahasa Indonesia. Hasil yang didapatkan menunjukan metode *Naïve Bayes* memiliki kinerja yang lebih baik dengan tingkat akurasi 70% sedangkan metode *K-Nearest Neighbor* memiliki tingkat akurasi yuang cukup rendah yaitu 40%.

Contoh selanjutnya adalah pada penelitian yang dilakukan oleh (Rahman, et al., 2017) dimana dalam penelitiannya, metode *Multinomial* *Naïve Bayes* digunakan untuk Klasifikasi Berita Online dengan menggunakan *feature* *selection* *Document Frequency Thresholding* dan menggunakan TF-IDF untuk pembobotan *term* dan menghasilkan akurasi tertinggi pada saat menggunakan TF-IDF 86,62%.

Tak hanya seleksi fitur, *stopword* merupakan salah satu tahapan penting dari *preprocessing*, dalam tahap *preprocessing* perlu adanya suatu mekanisme sistem daftar *stopword* dinamik yang dapat menghasilkan daftar *stopword* yang sesuai dengan yang diperlukan sesuai saran dari penelitian sebelumnya (Rahutomo & Ririd, 2018). Contoh pembuatan *stopword* dinamik ada pada penelitian selanjutnya yang dilakukan oleh (Dila Purnama Sari, et al., 2020) dimana dalam penelitiannya dilakukan pembentukan daftar *Stopword* menggunakan Zipf Law dan Pembobotan Augmented TF-Probability IDF pada klasifikasi Dokumen Ulasan Produk yang menggunakan metode Support Vector Machine dan Polynomial Kernel untuk memperoleh hasil klasifikasi. Daftar *Stopword* yang dibentuk secara dinamis dengan menggunakan metode Zipf Law dan pembobotan kata memiliki pengaruh terhadap hasil akurasi klasifikasi. Akurasi terbaik didapatkan pada saat persentase 15% untul daftar *stopword* yaitu dengan nilai *precision* 0,73, *recall* 0,7 dan *f-measure* 0,64

Lalu dilanjutkan penelitian oleh (Sa'rony, et al., 2019) analisis sentimen positif dan negatif yang dilakukan menggunakan *Multinomial* *Naïve Bayes* yang menggunakan Raw *Term* Frequency serta pembuatan *stopword* menggunakan *Term Based Random Sampling* dan berhasil mendapatkan *macroaverage* terbaik pada klasifikasi dengan stoplist 20 persen dengan *macroaverage* akurasi sebesar 0,94 *macroaverage* *precision* sebesar 0,945, *macroaverage* *recall* sebesar 0,94, dan *macroaverage* *f-measure* sebesar 0,938.

Contoh selanjutnya adalah pada penelitian (Imtiyazi, et al., 2015) dimana dilakukan perbandingan terhadap penggunaan *Multinomial* *Naïve Bayes* dengan TF-IDF dan dibandingkan terhadap *Multinomial* *Naïve Bayes* dengan TF-*Improved Gini*. Hasil yang didapatkan penggunaan TF-IDF memiliki kinerja lebih baik dibandingkan dengan TF-iGini.

Dari penelitian yang sudah disebutkan diatas, belum ada di antaranya yang melakukan Analisis Sentimen yang dibagi menjadi 3 kelas klasifikasi yaitu negatif, netral, dan positif yang menggunakan *Multinomial* *Naïve Bayes* sebagai metodenya serta pembentukan *Stopword* menggunakan Algoritma *Term Based Random Sampling* yang menggunakan TF-IDF sebagai pembobotan katanya.

### *New Normal*

Dengan adanya pandemi Covid-19 ini, pemerintah Indonesia mengadakan sistem *New Normal* yakni dengan tujun untuk mempercepat penangan COVID-19 (Putsanra, 2020).

### Kuliah Daring

Kuliah daring merupakan salah satu dari efek kebijakan sistem *New Normal* yang terjadi karena pandemi Covid-19 ini. Kuliah daring adalah metode pembelajaran yang dilakukan secara daring (*online*) dengan menggunakan berbagai fasilitas seperti platform *Zoom*, *Google Meet, Google Classroom*, situs pembelajaran universitas, dan lain- lain. Dengan adanya fasilitas- fasilitas tersebut, mahasiswa dan dosen tetap dapat berinteraksi satu sama lain layaknya kuliah secara tatap muka atau offline (Tania, 2020).

### Twitter

Twitter merupakan sosial media besutan Amerika Serikat yang diluncurkan pada tahun 2006. Twitter merupakan salah satu contoh dari sosial media yang banyak digunakan oleh masyarakat Indonesia yang digunakan sebagai sarana pertukaran informasi di dunia digital. Dalam pengunaannya, Twitter memberi istilah kepada pertukaran informasi tersebut dengan nama *Tweets,* yang mana *Tweets* adalah suatu teks atau kata yang dibatasi panjang nya hingga 280 karakter yang nanti akan di-*posting* dalam *platform* Twitter tersebut.

## Teks *Pre-processing*

Teks pre-processing merupakan langkah awal yang dilakukan dalam analisis sentimen untuk menyiapkan data yang berupa teks agar mudah untuk diproses nantinya (Gaddam, 2019). Teks pre-processing ini meliputi, case folding, cleaning, tokenisasi, *stopword* removal dan stemming.

### Case Folding

Case folding adalah suatu tahapan untuk menyeragamkan kalimat menjadi huruf kecil atau lowercase semua. Contohnya, jika ada kalimat “Saya suka bermain Komputer” menjadi “saya suka bermain komputer”.

### Cleaning

Cleaning adalah suatu tahapan pembersihan kalimat dari simbol-simbol, tanda baca, maupun angka. Contohnya, jika ada kalimat “Selamat pagi Adis, semoga harimu menyenangkan!” menjadi “Selamat pagi Adis semoga harimu menyenangkan”.

### Tokenisasi

Tokenisasi adalah suatu tahapan untuk memisahkan antar kata dari suatu kalimat sehingga kata-kata tersebut menjadi satu tidak tergabung dengan kata-kata lainnya (Gaddam, 2019). Contohnya, jika ada kalimat “saya sedang bermain gitar” menjadi [‘saya’, ‘sedang’, ‘bermain’, ‘gitar’].

### *Stopword* Removal

*Stopword* Removal adalah suatu tahapan untuk menghilangkan kata-kata yang kurang relevan berdasarkan kamus *stopword* yang digunakan (Gaddam, 2019). Kamus *Stopword* yang digunakan dalam penelitian ini adalah kamus *stopword* yang bersifat dinamis yang akan dibuat sesuai dengan kebutuhan sistem.

### Stemming

Stemming adalah suatu tahapan untuk mencari kata dasar dari suatu kata (Gaddam, 2019). Contohnya jika ada kata “bermain” menjadi main.

## *Term Based Random Sampling*

*Term Based Random Sampling* adalah suatu metode yang dapat digunakan untuk menghasilkan daftar *stopword* secara otomatis berdasarkan seberapa informatif kata tertentu (Lo, et al., 2005). Kita dapat mengetahui apakah kata tersebut *stopword* atau bukan dengan melihat kepentingannya, semakin tidak penting kata tersebut, maka kemungkinan menjadi *stopword*-nya lebih tinggi. Untuk mencari nilai kepentingan dari suatu *term* dapat dilakukan dengan perhitungan dengan rumus dari teori *Kullback-Leibler.* Dengan rumus tersebut kita dapat memberi bobot dari suatu term pada dokumen sampel. Berikut rumus dari *Kullback-Leibler* direpresentasikan dalam Persamaan 2.1.

|  |  |
| --- | --- |
|  | ( 2.1 ) |

Yang dimana dipresentasikan dalam Persamaan 2.2 dan dipresentasikan dalam Persamaan 2.3 adalah sebagai berikut.

|  |  |
| --- | --- |
|  | ( 2.2 ) |
|  | ( 2.3 ) |

Keterangan :

: bobot *term* t pada dokumen sampel

: frekuensi kueri *term* dalam dokumen sampel

: jumlah dari panjang dokumen sampel

: frekuensi kueri *term* dari keseluruhan dokumen

: total token dari keseluruhan dokumen

Dalam perhitungannya dilakukan pemilihan acak *term* dari keseluruhan *term*, lalu kemudian ambil dokumen yang mengandung *term* tersebut dan cari semua term dalam dokumen tersebut. Setiap term dalam dokumen tersebut akan dilakukan perhitungan bobot nya menggunakan *Kullback-Leibler*. Lalu setelah perhitungan bobotnya diambil sejumlah X *term* yang diurutkan dari bobot terendah yang dimana X adalah parameter yang dapat diubah-ubah nantinya. Dalam prosesnya pemilihan *term* acak ini dilakukan sebanyak Y kali dimana Y adalah sebuah parameter yang dapat diisi secara manual untuk mendapatkan hasil yang terbaik. Setelah melakukan proses yang dijelaskan sebelumnya dihitung rata-rata keseluruhan bobot yang didapat oleh *term* lalu diambil sejumlah L dimana L adalah parameter yang dapat diubah nantinya. L adalah parameter yang menentukan berapa jumlah daftar *stopword* yang ingin digunakan.

## Pembobotan Kata (TF-IDF)

Pada tahap ini dilakukan pembobotan kata yang mempresentasikan kata-kata tersebut untuk dilakukan perhitungan nantinya. Salah satu metode dalam *term* weighting yang sering digunakan adalah *Term Frequency – Inverse Document Frequency* (𝑡𝑓. 𝑖𝑑𝑓) (Jones, 2004). Metode TF-IDF adalah penggabungan dua metode untuk melakukan pembobotan kata. TF atau *Term Frequency* adalah frekuensi kemunculan *term* pada suatu dokumen dan IDF atau *Inverse Document Frequency* adalah perhitungan inverse terhadap frekuensi dokumen yang mengandung kata tersebut (Prabowo, et al., 2016).Berikut rumus yang digunakan untuk perhitungan tf-idf direpresentasikan pada Persamaan 2.4 dan Persamaan 2.5.

Berikut perhitungan nilai log tf :

|  |  |
| --- | --- |
|  | ( 2.4 ) |

Berikut perhitungan nilai idf :

|  |  |
| --- | --- |
| 𝑖𝑑𝑓(𝑡) = | ( 2.5 ) |

Keterangan :

: frekuensi *term* pada dokumen d

: nilai *Inverse Document Frequency* suatu *term* t

: total dokumen

: nilai *Document Frequency* suatu *term* t

## Algoritme *Naïve Bayes*

Algoritma *Naïve Bayes* yaitu algoritma klasifikasi *supervised* yang berbasis dengan teorema bayes dengan asumsi independensi tiap fitur (Sawla, 2018). Algoritma ini menggunakan metode probabilistik dan statistik.

Algoritma ini mencari probabilitas tertinggi untuk proses klasifikasi. Perhitungan Algoritma *Naïve Bayes* direpresentasikan pada Persamaan 2.6.

|  |  |
| --- | --- |
|  | ( 2.6 ) |

Keterangan :

: Posterior atau Probabilitas kelas diberikan dokumen

: Prior atau Probabilitas awal muncul kategori

: Likelihood

Dalam perhitungan likelihood atau *conditional probability* salah satu metodenya adalah menggunakan Multinomial. Perhitungan *conditional probability* dengan multinomial direpresentasikan pada Persamaan 2.7

|  |  |
| --- | --- |
|  | ( 2.7 ) |

Keterangan :

: Likelihood dalam kelas

: Jumlah kemunculan kata pada kategori

: Jumlah semua total kemunculan kata pada kategori

: Jumlah term unik atau fitur

## *Confusion Matrix*

Confusion Matrix adalah pengukur performa dari klasifikasi pembelajaran mesin (*Machine Learning*) (Narkhede, 2018). Confusion Matrix berisikan tabel untuk menampilkan hasil evaluasi yang didalamnya terdapat 2 kolom yaitu kelas hasil prediksi dan kelas sebenarnya.

Tabel 2.1 Confusion Matrix

|  |  |  |  |
| --- | --- | --- | --- |
|  | | Predicted | |
| Negatif | Positif |
| Actual | Negatif | TN | FP |
| Positif | FN | TP |

Keterangan :

* True Negative (TN) : jumlah dokumen yang predicted negatif dan actual negatif
* False Positive (FP) : jumlah dokumen yang predicted positif namun actual negatif
* False Negative (FN) : jumlah dokumen yang predicted negatif namun actual positif
* True Postive (TP) : jumlah dokumen yang predicted positif dan actual positif.

Fungsi dari confusion matrix untuk mempermudah evaluasi hasil klasifikasi untuk mencari *accuracy*, *precision*, *recall* dan *f-measure*. Berikut rumus *accuracy* direpresentasikan pada Persamaan 2.8, *recall* direpresentasikan pada Persamaan 2.9, *precision* direpresentasikan pada Persamaan 2.10, *f-measure* direpresentasikan pada Persamaan 2.11.

* *Accuracy* : kesesuaian nilai prediksi dengan nilai aktual

|  |  |
| --- | --- |
|  | ( 2.8 ) |

* *Recall* : jumlah banyak atau sedikitnya kesesuaian informasi yang didapatkan berdasarkan sudut pandang kelas atau label yang digunakan

|  |  |
| --- | --- |
|  | ( 2.9 ) |

* *Precision* : tingkat ketepatan antara informasi yang diminta

|  |  |
| --- | --- |
|  | ( 2.10 ) |

* *F-measure* : bobot harmonic mean pada *recall* dan *precision*

|  |  |
| --- | --- |
|  | ( 2.11 ) |

## *K-Fold Cross Validation*

*K-Fold* *Cross Validation* adalah suatu metode yang berfungsi untuk membagi data sebanyak K dengan ukuran yang sama atau hampir sama rata. Pada implementasinya pengujian *K-Fold* ini dilakukan dengan iterasi sebanyak K dimana pada setiap iterasinya data dibagi menjadi 2 tipe yaitu data latih dan data uji (Singh & Shukla, 2016). Berikut contoh illustrasi dari *K-Fold* *Cross Validation* yang ditunjukkan pada Gambar 2.1.



Gambar 2.1 Ilustrasi *K-Fold* *Cross Validation*

Sumber : <https://www.researchgate.net/figure/The-technique-of-KFold-cross-validation-illustrated-here-for-the-case-K-4-involves_fig10_278826818> (2015)

Berdasarkan Gambar 2.1 ditunjukkan bahwa tiap iterasi dibagi menjadi 2 tipe data yaitu yang berwarna hijau adalah data latih dan yang berwarna biru adalah data uji. Untuk menghitung nilai evaluasi akhir maka dihitung rata-rata dari evaluasi tiap iterasi (Neale, et al., 2019).

# METODOLOGI

Pada bab ini akan dijelaskan metodologi yang digunakan pada penelitian ini. Metodologi yang digunakan berupa tipe penelitian, strategi penelitian, subjek penelitian, lokasi penelitian, teknik pengumpulan data, peralatan pendukung, implementasi algoritme, analisis, serta kesimpulan.

## Tipe Penelitian

Tipe penelitian yang dilakukan adalah bersifat non-implementatif dengan menggunakan pendekatan analitik. Penelitian bertipe non-implementatif adalah penelitian yang menguji hubungan terhadap suatu kejadian yang kemudian akan di analisis. Sedangkan pendekatan analitik memiliki fungsi untuk menjelaskan hubungan suatu kejadian dengan suatu objek penelitian yang sedang diteliti.

## Strategi Penelitian

Strategi penelitian ini menggunakan studi kasus analisis sentimen masyarakat terhadap kuliah daring yang didapat dari Twitter. Data tersebut dilabeli manual oleh pakar lalu dibagi menjadi data latih dan data uji. Studi eksperimen berfokus kepada pengujian pada parameter X, Y, dan L pada *Term Based Random Sampling.*

## Subjek Penelitian

Subjek penelitian yang digunakan pada penelitian ini adalah pengguna Twitter yang membahas mengenai kuliah daring.

## Peralatan Pendukung

Peralatan pendukung yang digunakan pada penelitian ini adalah:

Tabel 3.1 Spesifikasi Hardware

|  |  |
| --- | --- |
| Spesifikasi | Keterangan |
| Laptop | Dell XPS 15 9575 |
| CPU | Core i7-8750G |
| GPU | NVIDIA GeForce GTX 1050 (4GB GDDR5) |
| RAM | 16 GB |
| Tipe Memori | DDR4 |
| SSD | 512GB SSD PCIe NVMe |

Tabel 3.2 Spesifikasi Software

|  |  |
| --- | --- |
| Jenis | Keterangan |
| *Operating System* | MacOS Catalina 10.15.4 |
| Bahasa Pemrograman | Python 3.7.7 |
| IDE | Visual Studio Code |
| Library | Sastrawi, Pandas, Numpy, Re, Math |

## Lokasi Penelitian

Lokasi penelitian ini bertempat di Laboratorium Komputasi Cerdas, Fakultas Ilmu Komputer Universitas Brawijaya.

## Teknik Pengumpulan Data

Teknik yang digunakan untuk pengumpulan data pada penelitian ini berasal dari Pengguna Twitter. Data diambil menggunakan *library* *Twint* yang berfungsi sebagai *data scrapper* Twittter untuk *Python*. Kata kunci yang digunakan pada saat pengumpulan data adalah “Kuliah Daring”. Pengumpulan data dilakukan dalam 3 bulan terhitung sejak Juni 2020 hingga Agustus 2020.

## Data Penelitian

Pada penelitian kali ini, data yang digunakan adalah *tweet* berbahasa Indonesia. Total dokumen yang akan diambil dari Twitter adalah 300 dokumen dimana dari 300 dokumen akan dibagi menjadi 270 data latih, dan 30 data uji. Proses klasifikasi akan dibagi menjadi 3 yaitu positif, netral dan negatif.

## Teknik Analisis Data

Teknik Analisis Data pada penelitian ini ditujukan untuk mengetahui kinerja dari sistem yang telah dibuat sesuai algoritma yang diajukan oleh peneliti. Tingkat kinerja sistem diperoleh dengan menggunakan *Confusion Matrix* dan nantinya hasil yang diterima akan dimasukan ke dalam tabel Confusion Matrix dan dicari nilai *precision, recall, accuracy, dan f-measure* pada tiap iterasi *fold* dalam *K-fold cross validation*.

## Implementasi Algoritme

Implementasi Algoritme diawali dengan melakukan pembuatan daftar *Stopword* yang proses nya diawali dengan *preprocessing* data latih dengan tujuan untuk merubah data berbentuk kumpulan dokumen menjadi *term* untuk dilakukan perhitungan algoritma *Term Based Random Sampling*. Setelah daftar *stopword* didapatkan, data latih akan melalui tahap *preprocessing* yang dimana didalamnya terdapat tahap-tahapan seperti *case folding, cleaning,* tokenisasi, *stopword removal* dengan menggunakan daftar *stopword* yang telah dibuat, lalu diakhiri dengan *stemming*. Lalu dilanjutkan dengan pembobotan kata menggunakan 𝑡𝑓. 𝑖𝑑𝑓 untuk merubah kata tersebut menjadi suatu nilai yang nantinya dapat diproses oleh sistem untuk dilatih dan diklasifikasi menggunakan metode *Multinomial* *Naïve Bayes*.

# Perancangan

Pada bab ini akan dijelaskan perancangan dengan diagram alir dari metode-metode yang digunakan pada penelitian ini serta manualisasi sistem klasifikasi dengan Naïve Bayes serta *Term Based Random Sampling* sebagai metode pembentuk daftar stopword.

## Diagram Alir Sistem

Pada diagram ini akan dijelaskan bagaimana tahapan-tahapan dari sistem. Tahapan-tahapan tersebut dijelaskan pada Gambar 4.1.

Mulai

*Preprocessing*

Data Uji, term, stopword, likelihood, prior

Mulai

*Preprocessing*

Data Latih, stopword

Mulai

Preprocessing tanpa Filtering

Data Latih

Term Based Random Sampling

*Term Weighting*

*Naïve Bayes Testing*

Naïve Bayes Training

Term, Likelihood, Prior

Hasil Klasifikasi  
Data Uji

Selesai

Selesai

Selesai

Daftar Stopword

|  |  |  |
| --- | --- | --- |
| (a) | (b) | (c) |

Gambar 4.1 Diagram Alir Sistem, (a) Tahap Pembuatan Stopword, (b) Tahap Pelatihan, (c) Tahap Pengujian

Berdasarkan pada Gambar 4.1 terdapat 3 diagram alir yaitu diagram tahap pembuatan stopword, tahap pelatihan, dan tahap pengujian. Pada tahap pembuatan stopword, masukan berupa data latih dan diawali dengan preprocessing data latih namun tidak menggunakan tahap filtering. Selanjutnya dilanjutkan oleh proses pembuatan daftar *stopword* menggunakan *Term Based Random Sampling* yang memiliki hasil berupa daftar stopword. Daftar stopword ini akan digunakan dalam tahap pelatihan dan tahap pengujian.

Pada tahap pelatihan, masukan berupa data latih dan stopword yang sudah dibuat sebelumnya dan diawali dengan preprocessing yang menggunakan stopword yang didapatkan sebagai masukan untuk proses filtering. Selanjutnya dilanjutkan oleh proses *Term Weighting* menggunakan metode *term frequency – inverse document frequency*. Setelah bobot didapatkan akan dilanjutkan proses pelatihan Naïve Bayes yang akan menghasilkan likelihood serta prior. Pada tahapan ini akan menghasilkan term, likelihood, prior yang akan digunakan pada tahap pelatihan.

Pada tahap pengujian, masukan berupa data uji, term, stopword, likelihood, dan juga prior. Tahapan ini diawali dengan preprocessing data uji menggunakan stopword yang sudah dibuat untuk proses filtering. Selanjutnya adalah Naïve Bayes Testing dimana didalam proses ini akan terjadi perhitungan posterior untuk masing-masing kategori. Sehingga akan ditentukan kelas klasifikasi dengan melihat posterior tertinggi.

### Diagram Alir *Preprocessing* tanpa Filtering

Pada tahapan *Preprocessing* tanpa *Filtering* ini terdapat beberapa tahapan yaitu *case folding, cleaning, tokenizing, stemming.* Preprocessing ini memiliki perbedaan dengan *preprocessing* pada umumnya karena tidak adanya *filtering* karena tujuan tahapan ini adalah proses menghasilkan suatu daftar *stopword.* Tahapan ini akan dijelaskan pada Gambar 4.2.

­

Mulai

Preprocessing tanpa Filtering

Data Latih

cleaned\_data =[],

terms=[]

i=0,

I < jumlah data latih

Case folding

Cleaning

Stemming

A

C

B

A

C

B

Tokenizing

Perulangan kata dalam tokenizing

Jika kata belum ada di terms

Kata masuk ke terms

Hasil Stemming ditambahkan ke cleaned\_data

Selesai

Keluaran

cleaned\_data, terms

Gambar 4.2 Diagram Alir Preprocessing tanpa Filtering

### Diagram Alir *Term Based Random Sampling*

Pada tahapan *Term Based Random Sampling* ini terdapat beberapa tahapan-tahapan untuk mendapatkan stopword berdasarkan dokumen tertentu. Tahapan ini akan dijelaskan pada Gambar 4.3.

Mulai

Term Based Random Sampling

Cleaned\_data,

terms, X, Y, L

A

token\_used = []

Tidak

A

i=0, i<Y

token\_w = {},  
normalized\_term\_w = {},  
sorted\_term\_w = {}

Ambil kata acak dari terms

Ambil dokumen dari data latih yang mengandung kata acak

Cari terms dari sampled\_documents

Perulangan word dalam term\_sampled\_documents

Kullback-Leibler Divergence

Jika word belum ada di token\_used

Hasil Kullback-Leibler Divergence dimasukan kedalam token\_w[word]

A

C

D

Ya

B

E

word dimasukan ke dalam token\_used

Ambil bobot maksimum dalam token\_w

Ambil bobot minimum dalam token\_w

Normalisasi MinMax Term Weighting

A

C

D

B

E

Urutkan normalized\_term\_weight dari terendah

Jika index lebih kecil dari X

Perulangan term\_weight dalam sort\_term\_weight

Term\_weight dimasukan ke sorted\_term\_weight

sorted\_term\_weight di tambahkan ke token\_weight

A

weighted\_token = {}

Ya

Tidak

temp = []

A

Perulangan token dalam token\_used

Perulangan tok\_w dalam token\_weight

Jika used\_token ada didalam tok\_w

tok\_w dengan key used\_tok akan ditambahkan ke temp

weighted\_token dengan key used\_tok diinsialisasi dengan temp

Ya

Tidak

merged\_weighted\_token = {}

Perulangan k,v dalam setiap item weighted\_token

Jika panjang v tidak sama dengan 0

merged\_weighted\_token dengan key k diinisialisasi dengan mean v

Urutkan merged\_weighted\_token dari rendah dan dimasukan kedalam variabel sorted\_merged\_weighted\_token

A

Ya

Tidak

Ambil term sejumlah L teratas dari sorted\_merged\_weighted\_token dan masukan ke dalam variabel sorted\_final\_weight

A

Perulangan k, v dalam setiap item sorted\_final\_weight

tok\_w dengan key used\_tok akan ditambahkan ke temp

stopwords = []

Keluaran stopwords

Selesai

Gambar 4.3 Diagram Alir *Term Based Random Sampling*

Pada tahapan Term Based Random Sampling ini terdapat beberapa tahapan yaitu diawali dengan pilih term acak dari keseluruhan term, ambil dokumen yang mengandung dokumen tersebut, hitung bobot tiap term menggunakan Kullback Leibler, normalisasi bobot dengan MinMax, ambil sejumlah X term yang diurutkan dari bobot terendah, lakukan proses sebelumnya sebanyak Y kali, hitung rata-rata keseluruhan bobot tiap term, dan yang terakhir ambil sejumlah L term.

#### Diagram Alir Kullback Leibler Divergence

Pada tahapan Kullback Leibler Divergence ini terdapat perhitungan pemberian bobot term untuk mendapatkan stopword berdasarkan dokumen tertentu. Tahapan ini akan dijelaskan pada Gambar 4.4.

Mulai

Kullback Leibler Divergence

word, sampled documents, cleaned\_data, terms

A

A

Hitung jumlah word dalam sampled\_documents dimasukan kedalam tf\_x

Hitung jumlah word dalam sampled\_documents

dimasukan ke dalam l\_x

Hitung jumlah word dalam cleaned\_data

dimasukan ke dalam F

Hitung jumlah token dalam keseluruhan dokumen dimasukan ke dalam token\_c

Keluaran w(t)

Mulai

Gambar 4.4 Diagram Alir Kullback Leibler Divergence

#### Diagram Alir Normalisasi MinMax Term Weighting

Pada tahapan ini terdapat perhitungan normalisasi bobot term dengan MinMax agar bobot dalam angka 0 hingga 1. Tahapan ini akan dijelaskan pada Gambar 4.5.

Mulai

Normalisasi MinMax Term Weighting

A

token\_w, max\_weight\_term, min\_weight\_term

A

normalized\_term\_weight = {}

Perulangan k,v dalam setiap item token\_w

Keluaran normalized\_term\_weight

Selesai

Hasil normalisasi akan dimasukan ke normalized\_term\_weight dengan key k

Gambar 4.5 Diagram Alir Normalisasi MinMax Term Weighting

### Diagram Alir *Preprocessing*

Tahapan *preprocessing* ini terdapat di 2 tahap yaitu tahap pelatihan dan tahap pengujian. Adapun tahap dalam preprocessing yaitu yaitu *case folding, cleaning, tokenizing, filtering, stemming.* Tahapan ini akan dijelaskan pada Gambar 4.6.

Mulai

Preprocessing

Data Latih, stopwords

cleaned\_data =[],

terms=[]

i=0,

I < jumlah data latih

A

C

B

A

C

B

*Tokenizing*

Perulangan kata dalam tokenizing

Jika kata belum ada di terms

Kata masuk ke terms

Hasil Stemming ditambahkan ke cleaned\_data

Selesai

Keluaran

cleaned\_data, terms

*Case folding*

*Cleaning*

*Filtering*

*Stemming*

Ya

Tidak

Gambar 4.6 Diagram Alir Preprocessing

### Diagram Alir *Term Weighting*

Tahapan *Term Weighting* ini terdapat beberapa tahapan-tahapan yaitu diawali dengan menghitung *raw term frequency*, *log term frequency, inverse document frequency,* dan *term frequency – inverse document frequency*. Tahapan ini akan dijelaskan pada Gambar 4.7.

Mulai

cleaned\_data, terms

*Raw Term Frequency*

*Log Term Frequency*

*Inverse Document Frequency*

*Term Frequency – Inverse Document Frequency*

Keluaran tf\_idf

Selesai

Term

Weighting

Gambar 4.7 Diagram Alir Term Weighting

#### Diagram Alir Raw Term Weighting

Tahapan *Raw Term Weighting* ini bertujuan untuk menghitung frekuensi setiap term yang terdapat dalam dokumen. Tahapan ini akan dijelaskan pada Gambar 4.8.

Mulai

*Raw Term Weighting*

cleaned\_data, terms

A

C

B

Perulangan term didalam terms

raw\_tf = {}

Keluaran raw\_tf

Selesai

A

C

B

temp = []

Perulangan data didalam cleaned\_data

Menghitung kemunculan term pada cleaned\_data dan dimasukan ke temp

raw\_tf dengan key term diinisialisasi dengan temp

Gambar 4.8 Diagram Alir *Raw Term Wighting*

#### Diagram Alir Log Term Weighting

Tahapan *Log Term Weighting* ini bertujuan untuk menghitung frekuensi setiap term yang terdapat dalam dokumen lalu di logaritma. Tahapan ini akan dijelaskan pada Gambar 4.9.

Mulai

*Log Term Weighting*

raw\_tf, terms

A

C

B

Perulangan term didalam terms

log\_tf = {}

Keluaran log\_tf

Selesai

A

C

B

temp = []

i = 0, i < panjang raw\_tf dengan key term

logtf dimasukkan ke temp

log\_tf[term] = temp

Jika raw\_tf key term dengan index i tidak sama dengan 0

Ya

Tidak

Gambar 4.9 Diagram Alir *Log Term Wighting*

#### Diagram Alir Inverse Document Frequency

Tahapan *Inverse Document Frequency* ini bertujuan untuk perhitungan inverse terhadap frekuensi dokumen yang mengandung kata tersebut. Tahapan ini akan dijelaskan pada Gambar 4.10.

Mulai

*Inverse Document Frequency*

A

raw\_tf, terms, cleaned\_data

idf = []

A

Perulangan term dalam terms

Hitung jumlah frekuensi dokumen yang memiliki term

Keluaran idf

Selesai

Hitung 𝑖𝑑𝑓(𝑡) =

Masukan hasil idf(t) ke list idf

Gambar 4.10 Diagram Alir *Inverse Document Frequency*

#### Diagram Alir Term Frequency - Inverse Document Frequency

Tahapan *Term Frequency - Inverse Document Frequency* ini bertujuan untuk mengkalikan *log term frequency* dengan *inverse document frequency*. Tahapan ini akan dijelaskan pada Gambar 4.11.

Mulai

*Term Frequency – Inverse Document Frequency*

terms, log\_tf, idf

Perulangan term didalam terms

tf\_idf = {}, count = 0

temp = []

i = 0, I < panjang dokumen

A

C

D

B

E

Hasil

log\_tf[term][i] \* idf[count] dimasukan ke temp

A

C

D

B

E

tf\_idf[term] = temp,

count+=1

Keluaran tf\_idf

Selesai

Gambar 4.11 Diagram Alir *Term Frequency - Inverse Document Frequency*

### Diagram Alir *Naïve Bayes Training*

Pada tahapan Naïve Bayes Training ini terdapat beberapa tahapan yaitu mencari likelihood setiap kelas serta mencari prior tiap kelasnya. Tahapan ini akan dijelaskan pada Gambar 4.12.

Mulai

*Term Frequency – Inverse Document Frequency*

cleaned\_data, terms, target, stopword, tf\_idf

i = 0, i < panjang cleaned\_data

total = []

total\_word = 0

Perulangan term didalam terms

total\_word += tf\_idf[term][i]

A

C

B

A

C

B

total\_word tambahkan ke total

con\_prob\_negative = [], con\_prob\_neutral = [],

con\_prob\_positive = []

Perulangan term didalam terms

Hitung Likelihood term tiap kelas

Likelihood\_negatif dimasukan ke con\_prob\_negatif

Likelihood\_ neutral dimasukan ke con\_prob\_neutral

Likelihood\_ positive dimasukan ke con\_prob\_positive

terms\_con\_prob = {},

i = 0

Perulangan term didalam terms

con\_prob\_negatif index i

ditambahkan ke temp

temp = []

con\_prob\_neutral index i

ditambahkan ke temp

con\_prob\_positive index i

ditambahkan ke temp

i +=1

A

A

Hitung Prior tiap kelas

Keluaran likelihood, prior, term

Selesai

Gambar 4.12 Diagram Alir *Naive Bayes Training*

#### Diagram Alir Hitung Likelihood term tiap kelas

Pada tahapan Hitung *Likelihood term* tiap Kelas ini yaitu mencari likelihood term tertentu setiap kelasnya. Tahapan ini akan dijelaskan pada Gambar 4.13.

Mulai

Hitung Likelihood term tiap kelas

kata, kategori

total = []

Hitung jumlah word didalam kategori dimasukan kedalam count\_w\_c

Hitung jumlah semua term didalam kategori dimasukan ke dalam count\_c

Hitung jumlah semua term dimasukan ke dalam V

Keluaran P(w|c)

Selesai

Gambar 4.13 Diagram Alir Hitung Likelihood term tiap kelas

#### Diagram Alir Hitung Prior tiap kelas

Pada tahapan Hitung *Prior* tiap Kelas ini yaitu mencari prior setiap kelasnya. Tahapan ini akan dijelaskan pada Gambar 4.14.

Mulai

Hitung Likelihood term tiap kelas

kelas

prior\_negatif, prior\_neutral, prior\_positive

Hitung total dokumen dikelas negatif dibagi total keseluruhan dokumen dan hasil dimasukan ke dalam prior\_negatif

Keluaran prior\_negatif, prior\_neutral, prior\_positive

Selesai

Hitung total dokumen dikelas negatif dibagi total keseluruhan dokumen dan hasil dimasukan ke dalam prior\_negatif

Hitung total dokumen dikelas negatif dibagi total keseluruhan dokumen dan hasil dimasukan ke dalam prior\_negatif

Gambar 4.14 Diagram Alir Hitung Prior tiap kelas

### Diagram Alir *Naïve Bayes Testing*

Pada tahapan Naïve Bayes Testing ini berfungsi untuk menghitung posterior setiap kelasnya dari data uji. Tahapan ini akan dijelaskan pada Gambar 4.15.

Mulai

*Naïve Bayes Testing*

terms, cleaned\_data\_test, terms\_test

A

A

used\_terms = []

Perulangan term dalam terms\_test

Jika term ada di terms

term ditambahkan ke used\_terms

used\_terms\_with\_likelihood = {}

Perulangan term dalam used\_terms

temp = []

likelihood[term][0]

dimasukkan ke temp

likelihood[term][1]

dimasukkan ke temp

likelihood[term][2]

dimasukkan ke temp

temp dimasukkan ke

used\_terms\_with\_likelihood[term]

negatif = 1,

netral = 1,

positif = 1

A

A

Perulangan term dalam used\_terms

negatif\*= used\_terms\_with\_likelihood[term][0]

netral\*= used\_terms\_with\_likelihood[term][1]

positif\*= used\_terms\_with\_likelihood[term][2]

negatif = negatif \* prior\_negative

netral = netral \* prior\_ neutral

positif = positif \* prior\_positive

Jika negatif lebih besar dari netral dan positif

Selain itu

Jika positif lebih besar dari negatif dan netral

final\_result = “Positif”

final\_result = “Netral”

final\_result = “Negatif”

Keluaran final\_result

Selesai

Ya

Ya

Ya

Tidak

Tidak

Gambar 4.15 Diagram Alir Naive Bayes Testing

# DAFTAR REFERENSI

Arnani, M. (2020, Maret 13). *KOMPAS*. Diambil kembali dari Kasus Pertama Virus Corona di China Dilacak hingga 17 November 2019: https://www.kompas.com/tren/read/2020/03/13/111245765/kasus-pertama-virus-corona-di-china-dilacak-hingga-17-november-2019

Ramadhan, A., Nugraheny, D. E., & Maharani, T. (2020, September 5). *KOMPAS*. Diambil kembali dari UPDATE: Kembali Bertambah di Atas 3.000, Kasus Covid-19 Lewati 190.000: https://nasional.kompas.com/read/2020/09/05/15204581/update-kembali-bertambah-di-atas-3000-kasus-covid-19-lewati-190000?page=all

Putsanra, D. V. (2020, Mei 26). *tirto*. Diambil kembali dari Apa itu New Normal dan Bagaimana Penerapannya Saat Pandemi Corona?: https://tirto.id/apa-itu-new-normal-dan-bagaimana-penerapannya-saat-pandemi-corona-fCSg

Liu, B. (2012). *Sentiment Analysis and Opinion Mining.* Chicago: Morgan & Claypool.

Sa'rony, A., Adikara, P. P., & Wihandika, R. C. (2019). Analisis Sentimen Kebijakan Pemindahan Ibukota Republik Indonesia dengan Menggunakan Algoritme Term-Based Random Samplingdan Metode Klasifikasi Naïve Bayes. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 10086-10094.

Dila Purnama Sari, D. E., Sari, Y. A., & Furqon, M. T. (2020). Pembentukan Daftar Stopword menggunakan Zipf Law dan Pembobotan Augmented TF - Probability IDF pada Klasifikasi Dokumen Ulasan Produk. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 406-412.

Rahutomo, F., & Ririd, A. T. (2018). EVALUASI DAFTAR STOPWORD BAHASA INDONESIA. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, 41-48.

Septian, J. A., Fahrudin, T. M., & Nugroho, A. (2019). Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor. *JOURNAL OF INTELLIGENT SYSTEMS AND COMPUTATION*.

Devita, R. N., Herwanto, H. W., & Wibawa, A. P. (2018). ERBANDINGANKINERJAMETODE NAIVE BAYES DAN K-NEAREST NEIGHBOR UNTUK KLASIFIKASI ARTIKEL BERBAHASA INDONESIA. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK), 5*(4), 427-434.

Rahman, A., Wiranto, & Doewes, A. (2017). Online News Classification Using Multinomial Naive Bayes. *ITSMART: Jurnal Ilmiah Teknologi dan Informasi*.

Imtiyazi, M. A., Shaufiah, & Bijaksana, M. A. (2015). Sentiment Analysis Berbahasa Indonesia Menggunakan Improved Multinomial Naive Bayes. *e-Proceeding of Engineering, 2*(2), 6331.

Gaddam, S. H. (2019, November 21). *Text Preprocessing in Natural Language Processing*. Diambil kembali dari towards data science: https://towardsdatascience.com/text-preprocessing-in-natural-language-processing-using-python-6113ff5decd8

Sawla, S. (2018, June 9). *Introduction to Naive Bayes for Classification* . Diambil kembali dari medium: https://medium.com/@srishtisawla/introduction-to-naive-bayes-for-classification-baefefb43a2d

Narkhede, S. (2018, May 9). *Understanding Confusion Matrix*. Diambil kembali dari towards data science: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Tania, A. (2020). *Muda Kompas*. Dipetik September 22, 2020, dari https://muda.kompas.id/baca/2020/05/13/perlu-kerjasama-dosen-dan-mahasiswa-dalam-kuliah-daring/

Lo, R. T.-W., He, B., & Ounis, I. (2005). *Automatically Building a Stopword List for an Information Retrieval System.* Glasgow, UK: Department of Computing Science.

Jones, S. (2004). A Statistical Interpretation of Term Specificity and Its Retrieval. *Journal Of Documentation, 60*(5), 11-21.

Prabowo, D. A., Fhadli, M., Najib, M. A., & Fauzi, H. A. (2016). TF-IDF-Enhanced Genetic Algorithm Untuk Extractive Automatic Text Summarization. *Jurnal Teknologi Informasi dan Ilmu Komputer, 3*(3), 208-215.

Singh, S., & Shukla, S. (2016). Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification. Bhimavaram: IEEE.

Neale, C., Workman, D., & Dommalapati, A. (2019). *Cross Validation: A Beginner’s Guide*. Dipetik September 23, 2020, dari https://towardsdatascience.com/cross-validation-a-beginners-guide-5b8ca04962cd