

**ANALISIS SENTIMEN PADA ULASAN APLIKASI MOBILE
MENGUNAKAN NAÏVE BAYES CLASSIFIER DAN ASOSIASI TEKS
(STUDI KASUS: APLIKASI PLN MOBILE)**

LAPORAN KERJA PRAKTIK



Disusun oleh
ALFIAN HIDAYATULLOH
5180411382

**PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS & TEKNOLOGI
UNIVERSITAS TEKNOLOGI YOGYAKARTA
YOGYAKARTA
2021**

LAPORAN KERJA PRAKTIK

**ANALISIS SENTIMEN PADA ULASAN APLIKASI MOBILE
MENGUNAKAN NAÏVE BAYES CLASSIFIER DAN ASOSIASI TEKS
(STUDI KASUS: APLIKASI PLN MOBILE)**

disusun oleh

ALFIAN HIDAYATULLOH

5180411382

Telah dipertanggungjawabkan di dalam Seminar Kerja Praktik

Pada tanggal 15 Juni 2021

Dosen Pembimbing



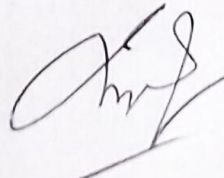
Rr. Hajar Puji Sejati, S.T., M.Kom.

NIK. 110918171

Yogyakarta, 20-06-2021

Ketua Program Studi Informatika

Fakultas Sains & Teknologi, Universitas Teknologi Yogyakarta



Dr. Enny Itje Sela, S.Si., M.Kom.

NIK. 111116086

LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini

Nama : Alfian Hidayatulloh

NPM : 5180411382

Program Studi : Informatika

menyatakan dengan sesungguhnya bahwa laporan kerja praktik ini merupakan hasil penelitian, pemikiran, dan pemaparan hasil saya sendiri. Saya tidak mencamtumkan tanpa pengakuan, bahan-bahan yang telah dipublikasikan sebelumnya, ditulis oleh orang lain, atau sebagai bahan yang pernah diajukan untuk gelar atau ijazah di Universitas Teknologi Yogyakarta maupun perguruan tinggi lainnya. Apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik sesuai dengan peraturan yang berlaku di Universitas Teknologi Yogyakarta. Demikian surat pernyataan ini saya buat dengan sadar dan dengan keterangan yang sebenarnya.

Yogyakarta, 18 Juni 2021

Yang menyatakan



Alfian Hidayatulloh

ABSTRAK

Perkembangan teknologi yang cepat ikut serta dalam memengaruhi perubahan gaya hidup masyarakat, salah satunya yaitu alat pembayaran. Pada era digitalisasi ini, banyak perusahaan yang mulai menerapkan alat pembayaran digital agar dapat terus bertahan setiap tahunnya, begitu juga yang dilakukan oleh PT PLN (Persero). Perusahaan ini telah beroperasi selama 76 tahun, tercatat hingga tahun 2019 memiliki jaringan distribusi sebanyak 1.042 Unit Layanan yang tersebar di seluruh wilayah Indonesia. PT PLN (Persero) juga memiliki sebuah aplikasi *mobile* pada situs Google Play Store, yang telah mendapatkan lebih dari 50.000 ulasan pengguna. Tidak bisa dipungkiri bahwa ulasan yang ditulis oleh pengguna dapat mempengaruhi citra sebuah perusahaan. Namun, memantau dan mengelola ulasan dari pengguna juga bukanlah hal yang mudah. Apalagi jika ulasan yang dimuat jumlahnya terlalu banyak untuk diproses secara manual. Oleh sebab itu, penulis melakukan penelitian menggunakan data ulasan pengguna aplikasi PLN Mobile dari situs Google Play Store. Jumlah ulasan pengguna pada penelitian ini berjumlah 3.200 data, yang kemudian dilakukan normalisasi, lalu dilabelisasi menjadi kelas sentimen positif dan negatif. Dari hasil labelisasi data, selanjutnya dilakukan proses klasifikasi menggunakan algoritma *Naïve Bayes Classifier* dan memperoleh tingkat akurasi tertinggi sebesar 90,88%. Pada penelitian ini juga dilakukan asosiasi teks untuk mengetahui informasi yang dianggap penting dan berguna dalam pengambilan keputusan. Asosiasi teks pada kelas sentimen positif adalah terkait aplikasi, transaksi, layanan, cepat, dan harga. Sedangkan pada kelas sentimen negatif adalah terkait mati, ganggu, padam, susah, dan *error*.

Kata Kunci: *Web Scraping*, Analisis Sentimen, *Naïve Bayes Classifier*, Asosiasi Teks, Google Play Store, PLN Mobile

ABSTRACT

Rapid technological developments have participated in influencing changes in people's lifestyles, one of which is a payment instrument. In this digitalization era, many companies are starting to implement digital payment instrument in order to continue to survive every year, as well as PT PLN (Persero). This company has been operating for 76 years, it was recorded that until 2019 it had a distribution network of 1,042 Service Units spread throughout Indonesia. PT PLN (Persero) also has a mobile application on the Google Play Store site, which has received more than 50,000 user reviews. It is undeniable that reviews written by users can affect the image of a company. However, monitoring and managing reviews from users is also not easy. Especially if the reviews that are loaded are too many to be processed manually. Therefore, the author conducted research using user reviews of the PLN Mobile application from the Google Play Store site. The number of user reviews in this study amounted to 3,200 data, which were then normalized, then labeled into positive and negative sentiment classes. From the results of data labeling, then the classification process is carried out using the Naïve Bayes Classifier algorithm and obtains the highest accuracy rate of 90.88%. In this study, text associations were also conducted to find out information that was considered important and useful in decision making. Text association in the positive sentiment class are related to application, transaction, service, speed, and price. Meanwhile, in the negative sentiment class, it is related to off, disturbance, blackout, difficulty, and error.

Keywords: *Web Scraping, Sentiment Analysis, Naïve Bayes Classifier, Text Association, Google Play Store, PLN Mobile*

KATA PENGANTAR

Alhamdulillah, segala puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat, taufik, dan hidayah-Nya sehingga penulis dapat menyusun dan menyelesaikan laporan kerja praktik yang berjudul Analisis Sentimen Pada Ulasan Aplikasi Mobile Menggunakan *Naïve Bayes Classifier* dan Asosiasi Teks (Studi Kasus: Aplikasi PLN Mobile) dengan baik dan lancar. Laporan ini ditulis sebagai salah satu persyaratan yang harus dipenuhi dalam menyelesaikan program sarjana di Jurusan Informatika, Fakultas Sains & Teknologi, Universitas Teknologi Yogyakarta. Dalam pelaksanaan dan penyusunan laporan, penulis banyak mendapatkan bantuan, dukungan, dan kesempatan dari berbagai pihak. Untuk itu, penulis ingin mengucapkan terima kasih kepada:

1. Dr. Bambang Moertono Setiawan, M.M., C.A., Akt. selaku Rektor Universitas Teknologi Yogyakarta.
2. Dr. Endy Marlina, S.T., M.T. selaku Dekan Fakultas Sains dan Teknologi, Universitas Teknologi Yogyakarta.
3. Dr. Enny Itje Sela, S.Si., M.Kom. selaku Ketua Program Studi Informatika, Universitas Teknologi Yogyakarta.
4. Rr. Hajar Puji Sejati, S.T., M.Kom. selaku Dosen Pembimbing Kerja Praktik yang telah memberi arahan dan nasihat kepada penulis.
5. Kedua orang tua yang telah memberi doa dan dukungan dalam menyelesaikan laporan kerja praktik ini.

Akhir kata, penulis berharap semoga laporan kerja praktik ini berguna bagi para pembaca dan pihak-pihak lain yang berkepentingan.

Yogyakarta, 18 Juni 2021

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	ii
LEMBAR PERNYATAAN	iii
ABSTRAK	iv
ABSTRACT	v
KATA PENGANTAR.....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR.....	x
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
1.6 Sistematika Penulisan	4
BAB II KAJIAN HASIL PENELITIAN DAN LANDASAN TEORI	6
2.1 Kajian Hasil Penelitian.....	6
2.2 Landasan Teori.....	9
2.2.1 Natural Language Processing.....	9
2.2.2 Web Scraping	10
2.2.3 Text Mining.....	11
2.2.4 Analisis Sentimen	12
2.2.5 Text Preprocessing	13
2.2.6 Pembobotan Kata	14
2.2.7 Teorema Bayes.....	16
2.2.8 Naïve Bayes Classifier	17
2.2.9 Confusion Matrix	20
2.2.10 Asosiasi Teks	21
BAB III METODE KERJA PRAKTIK	23
3.1 Gambaran Umum Instansi	23
3.1.1 Visi, Misi, dan Struktur Organisasi.....	24
3.1.2 Layanan dan Produk.....	28
3.1.3 Aturan Bisnis	33
3.2 Tahapan Penyelesaian Masalah.....	34
3.2.1 Identifikasi Masalah	34
3.2.2 Pengumpulan Data	35
3.2.3 Analisis Data	35
3.2.4 Perancangan Sistem	35
3.2.5 Implementasi Sistem	35

3.2.6 Pengujian Sistem.....	35
3.3 Bahan dan Data	36
3.3.1 Data yang Diperoleh	36
3.3.2 Prosedur Pengumpulan Data	37
BAB IV ANALISIS DAN DESAIN SISTEM	38
4.1 Analisis Sistem.....	38
4.1.1 Analisis Kebutuhan Fungsional	38
4.1.2 Analisis Kebutuhan Nonfungsional	38
4.2 Desain Sistem.....	39
4.2.1 Desain Logik	39
4.2.2 Desain Fisik.....	41
BAB V IMPLEMENTASI DAN HASIL	42
5.1 Implementasi	42
5.1.1 Proses Web Scraping.....	42
5.1.2 Proses Data Preprocessing	44
5.1.3 Proses Labelisasi Data	47
5.1.4 Proses Klasifikasi Sentimen.....	49
5.1.5 Proses Asosiasi Teks	52
5.2 Hasil	55
5.2.1 Hasil Proses Web Scraping	55
5.2.2 Hasil Proses Data Preprocessing	56
5.2.3 Hasil Proses Labelisasi Data	61
5.2.4 Hasil Proses Klasifikasi Sentimen.....	63
5.2.5 Hasil Proses Asosiasi Teks.....	70
BAB VI PENUTUP	77
6.1 Simpulan	77
6.2 Saran.....	78
DAFTAR PUSTAKA	79

DAFTAR TABEL

Tabel 2.1 Tabel perbandingan penelitian	7
Tabel 2.2 <i>Confusion matrix</i> untuk klasifikasi biner	20
Tabel 3.1 Contoh hasil <i>web scraping</i>	37
Tabel 5.1 Hasil proses <i>case folding</i>	57
Tabel 5.2 Hasil proses <i>cleaning</i> pertama.....	57
Tabel 5.3 Hasil proses <i>cleaning</i> kedua.....	58
Tabel 5.4 Hasil proses <i>cleaning</i> ketiga.....	58
Tabel 5.5 Hasil proses <i>cleaning</i> keempat	58
Tabel 5.6 Hasil proses <i>spelling normalization</i>	59
Tabel 5.7 Hasil proses <i>stemming</i>	60
Tabel 5.8 Hasil proses <i>filtering</i>	61
Tabel 5.9 Simulasi perhitungan skor sentimen.....	62
Tabel 5.10 Distribusi kelas sentimen hasil labelisasi data	63
Tabel 5.11 Hasil proses <i>oversampling</i>	63
Tabel 5.12 Data latih 70% dan data uji 30%	64
Tabel 5.13 Data latih 75% dan data uji 25%	64
Tabel 5.14 Data latih 80% dan data uji 20%	64
Tabel 5.15 Data latih 85% dan data uji 15%	64
Tabel 5.16 Data latih 90% dan data uji 10%	65
Tabel 5.17 Data latih untuk simulasi.....	65
Tabel 5.18 Dokumen matriks untuk simulasi.....	65
Tabel 5.19 Hasil perhitungan TF-IDF untuk dokumen matriks	66
Tabel 5.20 Hasil perhitungan TF-IDF untuk dokumen matriks	66
Tabel 5.21 Hasil perhitungan probabilitas kosakata kelas positif	67
Tabel 5.22 Hasil perhitungan probabilitas kosakata kelas negatif	68
Tabel 5.23 Model probabilitas tiap kelas sentimen	68
Tabel 5.24 Prediksi sentimen pada data uji	69
Tabel 5.25 Hasil <i>confusion matrix</i>	69
Tabel 5.26 Asosiasi kata pada kelas sentimen positif	72
Tabel 5.27 Asosiasi kata pada kelas sentimen negatif	75

DAFTAR GAMBAR

Gambar 1.1 Evolusi sistem pembayaran	1
Gambar 2.1 Kelompok data merah dan hijau	18
Gambar 2.2 Penambahan data baru pada kelompok data	18
Gambar 3.1 Denah kantor pusat PT PLN (Persero)	24
Gambar 3.2 Struktur organisasi kantor pusat PT PLN (Persero)	25
Gambar 3.3 Diagram alir penelitian	34
Gambar 3.4 Data ulasan aplikasi PLN Mobile	36
Gambar 4.1 Blok diagram alur sistem	40
Gambar 4.2 <i>Flowchart</i> sistem	41
Gambar 4.3 Antarmuka <i>software</i> RStudio	41
Gambar 5.1 <i>Source code</i> proses <i>web scraping</i>	43
Gambar 5.2 <i>Source code</i> proses <i>web scraping</i> (lanjutan)	43
Gambar 5.3 <i>Source code</i> proses <i>data preprocessing</i> pertama.....	45
Gambar 5.4 <i>Source code</i> proses <i>data preprocessing</i> kedua	46
Gambar 5.5 <i>Source code</i> proses <i>data preprocessing</i> ketiga.....	46
Gambar 5.6 <i>Source code</i> proses labelisasi data.....	48
Gambar 5.7 <i>Source code</i> proses labelisasi data (lanjutan)	49
Gambar 5.8 <i>Source code</i> proses klasifikasi sentimen pertama.....	50
Gambar 5.9 <i>Source code</i> proses klasifikasi sentimen kedua.....	51
Gambar 5.10 <i>Source code</i> proses klasifikasi sentimen ketiga.....	52
Gambar 5.11 <i>Source code</i> proses asosiasi teks pertama.....	53
Gambar 5.12 <i>Source code</i> proses asosiasi teks kedua.....	54
Gambar 5.13 <i>Source code</i> proses asosiasi teks ketiga.....	54
Gambar 5.14 Hasil proses <i>web scraping</i>	56
Gambar 5.15 Hasil proses <i>tokenizing</i>	60
Gambar 5.16 Kata yang paling banyak muncul pada kelas positif	71
Gambar 5.17 <i>Wordcloud</i> ulasan positif	71
Gambar 5.18 Kata yang paling banyak muncul pada kelas negatif.....	74
Gambar 5.19 <i>Wordcloud</i> ulasan negatif	74

BAB I

PENDAHULUAN

1.1 Latar Belakang

Alat pembayaran di Indonesia berkembang sangat pesat dan maju. Alat pembayaran terus berkembang, seperti terlihat pada Gambar 1.1, dari tunai ke nontunai seperti alat pembayaran berbasis kertas (*paper-based*), misalnya cek dan giro. Selain itu, dikenal juga alat pembayaran *paperless* seperti transfer dana elektronik, serta alat pembayaran memakai kartu ATM, kartu kredit, kartu debit dan kartu prabayar (*card-based*). Pada satu dekade terakhir, telah terjadi gelombang digitalisasi ke kehidupan masyarakat yang mengubah perilaku masyarakat sehari-hari. Alat pembayaran pun semakin bervariasi dengan kehadiran uang elektronik berbasis peladen (*server-based*). Pola konsumsi masyarakat pun mulai bergeser dan menuntut pembayaran serba cepat dan aman melalui berbagai platform antara lain situs web, aplikasi *mobile*, dan kode QR. Adanya pembayaran digital diharapkan mampu mendorong efisiensi kegiatan ekonomi masyarakat ke depannya.



Gambar 1.1 Evolusi sistem pembayaran
(Sumber: Situs web Bank Indonesia)

Salah satu aplikasi pembayaran digital yang populer di Indonesia menurut data Google Play Store per April 2021 adalah PLN Mobile. Hingga saat ini, aplikasi PLN Mobile telah diunduh lebih dari 5 juta pengguna dan memiliki lebih dari 50 ribu ulasan pengguna. Ulasan yang diberikan oleh pengguna mencakup bukan hanya mengenai fitur pada aplikasi, namun juga pelayanan PT PLN (Persero) secara keseluruhan. Ulasan ini mencakup keluhan yang bersifat negatif dan saran yang bersifat positif. Ulasan pengguna merupakan salah satu media yang efektif dan efisien untuk menemukan informasi terhadap citra dari suatu perusahaan. Hal ini dikarenakan konsumen melihat perspektif rekomendasi *word-of-mouth* pengguna lain sebelum menggunakan suatu produk maupun jasa (Fanani, F., 2017).

Google Play Store merupakan layanan digital milik Google berupa toko untuk memasarkan produk berupa aplikasi, permainan, hingga buku. Google Play Store dapat diakses melalui aplikasi Android, situs web, dan Google TV (Marziah, K., 2020). Google Play memiliki beragam fitur, salah satunya adalah pengguna dapat memberikan ulasan terhadap aplikasi. Berdasarkan ulasan tersebut dapat diketahui sentimen apa saja yang mempengaruhi dan perlu diperbaiki oleh perusahaan. Ulasan dapat dianalisis menggunakan metode *text mining*. *Text mining* adalah proses penemuan pengetahuan dengan mengekstrak pola atau pengetahuan yang dianggap penting dari data teks (Bookhamer, P. dan Zhang, Z.J., 2016). Salah satu teknik dari *text mining* adalah analisis sentimen yang mana dapat digunakan untuk mengklasifikasikan opini masyarakat ke dalam kelas positif, negatif, maupun netral (Suprpto, F., 2015). Apabila ditemukan adanya ulasan pada kelas sentimen negatif, maka PT PLN (Persero) dapat dengan cepat mengambil tindakan untuk mengatasi dan menyelesaikan keluhan pengguna.

Dalam penelitian ini, klasifikasi sentimen akan dilakukan menggunakan algoritma *Naïve Bayes Classifier*. *Naïve Bayes Classifier* adalah algoritma yang memberikan manfaat perhitungan statistik untuk memprediksi probabilitas di masa depan berdasarkan pengalaman di masa sebelumnya (Pamungkas, A., 2016). Kelebihan dari algoritma *Naïve Bayes Classifier* dalam proses klasifikasi adalah dalam jumlah data latih yang diperlukan untuk menentukan estimasi parameter. Jumlah tersebut cukup kecil dan menggunakan performa kerja yang jauh lebih

efisien dibandingkan algoritma klasifikasi lainnya (Pattekari, S.A. dan Parveen, A., 2012). Setelah melakukan klasifikasi, dilakukan proses eksplorasi informasi seluas-luasnya dari masing-masing kelas sentimen positif dan negatif. Proses eksplorasi yang dilakukan menggunakan asosiasi teks untuk menemukan topik-topik yang umumnya dibahas oleh pengguna beserta keterkaitan antartopik tersebut.

Dari latar belakang tersebut, penelitian ini akan menganalisis sentimen ulasan aplikasi *mobile* dengan objek penelitian adalah PLN Mobile. Data ulasan diambil dari kolom ulasan di Google Play Store. Melalui penelitian ini, diharapkan nantinya dapat memberikan informasi yang berguna untuk PT PLN (Persero) secara khusus maupun pihak lain yang membutuhkan.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan, maka rumusan masalah dalam penelitian ini meliputi:

1. Apakah algoritma *Naïve Bayes Classifier* mampu dalam mengklasifikasikan ulasan aplikasi PLN Mobile menjadi kelas sentimen positif dan negatif?
2. Bagaimana akurasi algoritma *Naïve Bayes Classifier* dalam mengklasifikasikan ulasan aplikasi PLN Mobile?
3. Apa saja informasi penting yang diperoleh dari hasil asosiasi teks pada ulasan aplikasi PLN Mobile?

1.3 Batasan Masalah

Batasan masalah yang ditentukan untuk menghindari perluasan pembahasan dalam penelitian ini adalah sebagai berikut:

1. Data ulasan yang digunakan berasal dari Google Play Store.
2. Data ulasan yang digunakan adalah ulasan (teks) berbahasa Indonesia.
3. Data ulasan yang tidak dapat dinormalisasi menggunakan perangkat lunak, dilakukan normalisasi secara manual oleh penulis berdasarkan acuan dari Kamus Besar Bahasa Indonesia (KBBI) dan Pedoman Umum Ejaan Bahasa Indonesia (PUEBI).

4. Setelah dinormalisasi, data ulasan akan diklasifikasi dan diasosiasi menjadi dua kelas sentimen, yakni positif dan negatif.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah mengetahui sentimen dari ulasan pengguna aplikasi PLN Mobile di Google Play Store dan memperoleh informasi penting yang dapat dimanfaatkan dalam pengambilan keputusan perusahaan untuk mengevaluasi pengembangan aplikasi PLN Mobile.

1.5 Manfaat Penelitian

Pada penelitian ini, terdapat beberapa manfaat yang akan diperoleh berdasarkan tujuan penelitian, antara lain:

1. Bagi penulis, menerapkan ilmu pengetahuan yang telah diperoleh selama perkuliahan dan mengetahui implementasi algoritma *Naïve Bayes Classifier* dalam melakukan klasifikasi data ulasan aplikasi PLN Mobile.
2. Bagi lingkup akademis, dapat dijadikan contoh studi kasus, acuan, serta referensi untuk melakukan penelitian terkait di masa mendatang.
3. Bagi instansi, dapat dijadikan sebagai wawasan dalam mengambil keputusan bagi PT PLN (Persero) maupun pihak lain yang membutuhkan.

1.6 Sistematika Penulisan

Penulisan ini dibagi menjadi lima bab dan setiap bab terdiri dari beberapa sub-bab yang saling berhubungan satu dengan yang lainnya. Sistematika penulisan dalam penelitian ini mencakup hal-hal sebagai berikut:

BAB I PENDAHULUAN

Bab ini memberikan penjelasan terkait latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penelitian.

BAB II KAJIAN HASIL PENELITIAN DAN LANDASAN TEORI

Pada bab ini, akan dijelaskan tentang referensi-referensi yang terkait dan penjelasan teori-teori tentang analisis sentimen, yaitu algoritma *Naïve Bayes Classifier* dan asosisasi teks.

BAB III METODE KERJA PRAKTIK

Bab ini memuat keterangan yang berkaitan dengan institusi tempat kerja praktik, meliputi deskripsi umum instansi, visi misi instansi, produk atau jasa yang dihasilkan oleh instansi, alur bisnis instansi, permasalahan yang dihadapi oleh instansi dan tahap penyelesaiannya, metode pengumpulan data, serta perangkat-perangkat yang digunakan dalam mengerjakan laporan kerja praktik ini.

BAB IV IMPLEMENTASI DAN HASIL

Pada bagian implementasi dan hasil ini, akan dijelaskan analisis terhadap hasil pengumpulan data dan penguraian hasil pengolahan data. Hasil penelitian ini yaitu untuk mengetahui keakuratan dari klasifikasi menggunakan algoritma *Naïve Bayes Classifier* dan asosiasi teks berdasarkan keterkaitan kata pada kelas sentimen.

BAB V PENUTUP

Pada bab ini, dijelaskan mengenai kesimpulan yang diperoleh dari serangkaian pengerjaan kerja praktik yang telah dilakukan, serta memberikan beberapa saran sebagai pengembangan terhadap penelitian pada topik analisis sentimen.

BAB II

KAJIAN HASIL PENELITIAN DAN LANDASAN TEORI

2.1 Kajian Hasil Penelitian

Dalam melakukan penelitian ini, peneliti mengacu pada beberapa penelitian yang telah dilakukan sebelumnya, antara lain penelitian oleh Gunawan, F. dkk. (2017). Penelitian tersebut membahas analisis sentimen aplikasi BCA Mobile menggunakan normalisasi kata berbasis *Levenshtein Distance* dan algoritma *Naïve Bayes Classifier*. Normalisasi kata digunakan pada penelitian untuk membetulkan pola kata yang salah eja, yang dapat menyulitkan mesin dalam mengklasifikasikan ulasan ke dalam sentimen positif dan negatif. Hasil penelitian menunjukkan bahwa akurasi tertinggi yang diperoleh menggunakan algoritma *Naïve Bayes Classifier* tanpa dan dengan normalisasi kata berbasis *Levenshtein Distance*, yaitu 94,4% dan 96,9%.

Dalam penelitian yang dilakukan oleh Simanjuntak, R.A. (2018) menguraikan analisis sentimen aplikasi Gojek menggunakan teknik pembobotan *Term Frequency* dan *Multinomial Naïve Bayes*. Setelah dilakukan perhitungan frekuensi kemunculan kata, data akan diklasifikasikan ke dalam kelas sentimen positif, negatif, dan netral. Hasil penelitian menunjukkan bahwa akurasi dan *f-score* yang didapatkan menggunakan pembobotan *Term Frequency* dan algoritma *Naïve Bayes Classifier*, yaitu 95,2% dan 92,3%.

Penelitian lainnya yang menjadi acuan adalah penelitian yang dilakukan oleh Sari, A.E. dkk. (2019). Penelitian tersebut menggunakan seleksi fitur *Information Gain* dan algoritma *Naïve Bayes Classification* untuk mengklasifikasikan ulasan aplikasi Mandiri Online ke dalam kelas sentimen positif dan negatif, serta mengkategorikan berdasarkan factor kualitas perangkat lunak ISO/IEC 25010. Hasil penelitian menunjukkan bahwa akurasi tertinggi yang didapatkan menggunakan algoritma *Naïve Bayes Classifier* tanpa dan dengan seleksi fitur *Information Gain*, yaitu 93,33% dan 95,33%.

Mengacu pada penelitian lain yang dilakukan oleh Chohan, S. dkk. (2020). Penelitian tersebut berisi pemaparan tentang analisis sentimen aplikasi DuoLingo menggunakan teknik optimasi *Syntetic Minority Over-sampling Technique* (SMOTE) dan algoritma *Naïve Bayes Classifier*. Teknik optimasi dilakukan pada penelitian untuk mengatasi data dalam kelas sentimen dengan jumlah yang tidak seimbang. Hasil penelitian menunjukkan bahwa akurasi tertinggi yang diperoleh menggunakan algoritma *Naïve Bayes Classifier* tanpa dan dengan teknik optimasi *Syntetic Minority Over-sampling Technique* (SMOTE), yaitu 71,14% dan 91,95. Pada Tabel 2.1 berikut menjadi perbandingan penelitian sebelumnya dan penelitian yang akan dilakukan oleh penulis.

Tabel 2.1 Tabel perbandingan penelitian

No.	Judul Penelitian	Penulis	Kesimpulan
1.	Analisis Sentimen Pada Ulasan Aplikasi Mobile Menggunakan <i>Naïve Bayes</i> dan Normalisasi Kata Berbasis <i>Levenshtein Distance</i> (Studi Kasus: Aplikasi BCA Mobile)	Ferly Gunawan, M. Ali Fauzi, dan Putra Pandu Adikara	Metode klasifikasi <i>Naïve Bayes</i> dan <i>Levenshtein Distance</i> dapat diterapkan pada analisis sentimen ulasan aplikasi <i>mobile</i> . Penggunaan metode <i>Levenshtein Distance</i> untuk normalisasi kata yang mengalami kesalahan penulisan mampu meningkatkan hasil klasifikasi dengan nilai akurasi sebesar 96,9% dibandingkan hanya menggunakan klasifikasi <i>Naïve Bayes</i> dengan nilai akurasi 94,4%.

No.	Judul Penelitian	Penulis	Kesimpulan
2.	Analisis Sentimen Pada Layanan Gojek Indonesia Menggunakan <i>Multinomial Naïve Bayes</i>	Rosi Alfionita Simanjuntak	Algoritma <i>Multinomial Naïve Bayes</i> mampu mengklasifikasi sentimen <i>tweet</i> ke dalam kelas negatif, positif dan netral. Pengujian analisis sentimen pada layanan Gojek Indonesia dengan metode <i>Multinomial Naïve Bayes</i> menghasilkan akurasi <i>f-score</i> sebesar 92,3%.
3.	Klasifikasi Ulasan Pengguna Aplikasi Mandiri Online di Google Play Store dengan Menggunakan Metode <i>Information Gain</i> dan <i>Naïve Bayes Classifier</i>	Amalia Elma Sari, Sri Widowati, dan Kemas Muslim Lhaksmana	Klasifikasi sentimen menggunakan <i>Naïve Bayes Classifier</i> dengan <i>feature selection</i> <i>Information Gain</i> akan menghasilkan akurasi dan <i>f-measure</i> lebih baik daripada tanpa <i>feature selection</i> <i>Information Gain</i> , yaitu sebesar 91,33% dan 90,37% untuk ulasan Mandiri Online karena fitur-fitur diseleksi terlebih dahulu sebelum proses klasifikasi
4.	Analisis Sentimen Aplikasi Duolingo Menggunakan Metode <i>Naïve Bayes</i> dan <i>Synthetic Minority Over Sampling Technique</i>	Saifurrachman Chohan, Arifin Nugroho, Achmad Maezar Bayu Aji, dan Windu Gata	Penggunaan <i>Synthetic Minority Over Sampling Technique</i> dapat meningkatkan nilai dari akurasi dan AUC. Dan peningkatan yang dihasilkan cukup signifikan dari 77.14% untuk akurasi dan 0.526 untuk AUC pada pemodelan <i>Naïve Bayes</i> tanpa SMOTE, menjadi 91.95% untuk akurasi dan 0.740 untuk AUC pada pemodelan <i>Naïve Bayes</i> dengan SMOTE.

2.2 Landasan Teori

2.2.1 Natural Language Processing

Natural Language Processing (NLP) adalah bidang ilmu komputer dan linguistik berkaitan dengan interaksi antara komputer dan bahasa (alami) manusia (Liu, B., 2010). Artinya pada sistem, bahasa alami mencakup percakapan informasi dari basis data komputer ke dalam bahasa yang dapat dibaca oleh manusia. Sebuah sistem NLP harus memperhatikan pengetahuan terhadap bahasa itu sendiri, baik dari segi kata yang digunakan, bagaimana kata-kata tersebut digabung untuk menghasilkan suatu kalimat, apa arti dari sebuah kata, apa fungsi sebuah kata dalam sebuah kalimat dan sebagainya. Pengolahan bahasa alami mengenal beberapa tingkat pengolahan, antara lain:

a) Fonetik dan fonologi

Fonetik dan fonologi berhubungan dengan suara yang menghasilkan kata yang dapat dikenali. Proses pengolahan ini menjadi penting dalam aplikasi yang memakai metode *speech-based system*.

b) Morfologi

Morfologi merupakan pengetahuan tentang kata dan bentuknya yang dimanfaatkan untuk membedakan satu kata dengan kata lainnya. Pada tingkat ini juga dapat dipisahkan antara kata dan elemen lainnya. Misalnya kata “mengerjakan” yang dibagi menjadi “kerja” (kata dasar); me- (imbuhan depan); dan -kan (imbuhan belakang).

c) Sintaksis

Sintaksis merupakan pemahaman tentang urutan kata dalam pembentukan kalimat dan hubungan antarkata tersebut dalam proses perubahan bentuk dari kalimat menjadi bentuk yang sistematis. Pada tingkat ini dilakukan proses pengaturan tata letak suatu kata untuk membentuk kalimat yang dapat dikenali.

d) Semantik

Semantik merupakan pemetaan bentuk struktur sintaks dengan memanfaatkan tiap kata ke dalam bentuk yang lebih mendasar dan tidak tergantung struktur kalimat. Semantik mempelajari arti suatu kata dan

bagaimana arti dari kata-kata tersebut membentuk suatu arti kalimat yang utuh. Dalam tingkatan ini belum tercakup konteks dari kalimat tersebut.

e) **Pragmatik**

Pengetahuan pada tingkatan pragmatik berkaitan dengan masing-masing konteks yang berbeda tergantung pada situasi dan tujuan pembuatan sistem.

f) *Discourse knowledge*

Discourse knowledge melakukan pengenalan apakah suatu kalimat yang sudah dibaca dan dikenali sebelumnya sehingga dapat mempengaruhi arti dari kalimat selanjutnya. Informasi ini penting diketahui untuk melakukan pengolahan arti terhadap kata ganti orang dan untuk mengartikan aspek sementara dari informasi.

g) *Word knowledge*

Word knowledge mencakup arti sebuah kata secara umum dan apakah ada arti khusus bagi suatu kata dalam suatu percakapan dalam konteks tertentu.

2.2.2 Web Scraping

Website merupakan aplikasi yang di dalamnya terdapat berbagai dokumen multimedia (teks, gambar, animasi, maupun video) yang menggunakan protokol HTTP (*HyperText Transfer Protocol*), dan untuk mengaksesnya digunakan perangkat lunak yang disebut *browser*. Sedangkan *web scraping* adalah proses pengambilan sebuah dokumen semiterstruktur dari internet, umumnya berupa halaman web dalam bahasa markup seperti HTML (*HyperText Markup Language*) atau XHTML (*Extensible HyperText Markup Language*), dan menganalisis dokumen tersebut untuk diambil data tertentu dari halaman tersebut yang mana digunakan dalam kepentingan tertentu (Turland, M., 2010). Terdapat serangkaian langkah yang perlu dilakukan untuk melakukan *web scraping* (Josi, A. dkk., 2014) sebagai berikut:

- a) *Create scraping template*: Pembuatan kode program yang akan mempelajari dokumen HTML dari *website* yang akan diambil informasinya.

- b) *Explore site navigation*: Pembuatan kode program yang akan mempelajari teknik navigasi pada *website* yang akan diambil informasinya untuk ditirukan pada aplikasi *web scraper*.
- c) *Automate navigation and extraction*: Berdasarkan informasi yang didapatkan dari langkah 1 dan 2 diatas, aplikasi *web scraper* dibuat agar pengambilan informasi pada *website* dapat dilakukan secara otomatis.
- d) *Extract data and package history*: Informasi yang telah diperoleh dari langkah 3 akan disimpan dalam format *file* tertentu, dan untuk selanjutnya dikonversi ke dalam bentuk format data yang siap diolah.

2.2.3 Text Mining

Text mining merupakan proses penggalian informasi dari sekumpulan dokumen data berupa teks yang mengandung informasi yang tidak terstruktur dengan menggunakan analisis tertentu (Feldman, R. dan Sanger, J., 2007). Pekerjaan yang dilakukan dalam konsep *text mining* secara garis besar adalah penggalian deskriptif (*descriptive mining*) dan penggalian prediktif (*predictive mining*). Pekerjaan *predictive mining* meliputi klasifikasi dokumen ke dalam kategori-kategori, lalu menggunakan informasi tersebut untuk membuat keputusan. Misalnya, kepuasan pelanggan terhadap suatu produk dapat diketahui melalui dokumen komentar pembelian, sehingga komentar pelanggan di pembelian yang akan datang dapat diprediksi. Lalu dilakukan *descriptive mining* yang membantu perusahaan untuk melakukan pengelompokkan dokumen berdasarkan konsep yang telah ditentukan (Siraj, F. dan Abdoulha, M.A., 2007). Untuk memperoleh informasi akhir yang berguna bagi pemilik data, *text mining* harus melalui tiga tahap (Miner, G. dkk., 2012) sebagai berikut:

a) *Preprocessing*

Tahap ini mengkonversi informasi dari format yang belum terstruktur menjadi format yang dapat diproses.

b) Penyusun vektor

Untuk dapat dipahami oleh sistem operasi *text mining*, sebuah vektor representasi dari token-token kata perlu dibuat berdasarkan kemunculan kata tersebut dalam dokumen.

c) Ekstraksi informasi

Metode ekstraksi informasi yang digunakan dalam penelitian ini adalah klasifikasi yang membagi objek ke dalam kategori yang telah ditentukan (*supervised method*). Pendekatan model klasifikasi yang digunakan adalah teknik statistik atau *machine learning*. Pendekatan *machine learning* akan dipakai dalam penelitian ini di mana mesin akan mengoperasikan model yang mampu belajar dari contoh dokumen yang terklasifikasi.

2.2.4 Analisis Sentimen

Analisis sentimen merupakan proses memperoleh informasi dengan cara memahami, mengekstrak, dan mengolah data tekstual secara otomatis (Pang, B. dan Lee, L., 2008). Analisis sentimen mulai terkenal pada tahun 2013 sebagai salah satu cabang *text mining*, yang juga dikenal dengan *opinion mining*. Pada dasarnya, analisis sentimen digunakan untuk mengetahui tanggapan dan sikap dari suatu kelompok atau individu terhadap suatu topik bahasan kontekstual dalam keseluruhan dokumen. Tanggapan dan sikap tersebut dapat berupa pendapat atau penilaian atau evaluasi (teori appraisal), keadaan afektif (keadaan emosional penulis saat menulis), atau komunikasi emosional (efek emosional yang sampai pada pembaca) (Saraswati, N.W.S., 2013).

Secara umum domain yang sering membutuhkan analisis sentimen antara lain: produk konsumen, layanan dan jasa, serta peristiwa-peristiwa sosial dan politik yang memerlukan opini publik. Analisis sentimen memiliki kecenderungan terhadap penelitian mengenai pernyataan suatu pendapat yang mempunyai suatu sentimen, baik positif atau negatif. Oleh karenanya, pendapat memiliki pengaruh yang tinggi kepada perilaku seseorang, maka dapat dikatakan semua aktivitas seseorang terwakili dari pendapat tersebut. Hal ini terlihat dalam proses pengambilan keputusan di mana umumnya diambil dari pendapat orang-orang.

Pada sektor bisnis dan organisasi, pendapat dan opini publik menjadi sangat penting terhadap penilaian suatu produk dan jasa (Liu, B., 2012).

2.2.5 Text Preprocessing

Dalam proses *text mining*, dokumen yang digunakan harus dipersiapkan terlebih dahulu sebelum dapat digunakan dalam proses utama. Proses mempersiapkan dokumen atau dataset mentah disebut juga dengan proses *text preprocessing*. *Text preprocessing* berfungsi untuk mengubah data teks yang tidak terstruktur atau sembarang menjadi data yang terstruktur. Adapun serangkaian proses yang dilakukan dalam tahapan *text preprocessing* (Krouska, A. dkk., 2016) adalah sebagai berikut.

a) *Spelling normalization*

Spelling normalization adalah proses substitusi atau perbaikan kata-kata singkatan atau salah ejaan. Substitusi kata dilakukan untuk menghindari jumlah perhitungan dimensi kata yang melebar. Perhitungan dimensi kata akan melebar jika kata yang salah eja tidak diubah karena kata tersebut sebenarnya mempunyai maksud dan arti yang sama, tapi dianggap sebagai entitas yang berbeda pada saat proses penyusunan matriks.

b) *Case folding*

Case folding adalah proses penyamaan *case* (format huruf) dalam sebuah dokumen. Hal ini dilakukan untuk mempermudah pencarian. Tidak semua dokumen teks konsisten dalam penggunaan huruf kapital. Oleh karena itu, peran *case folding* dibutuhkan dalam mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar.

c) *Tokenizing*

Tokenizing atau tokenisasi adalah proses penguraian teks yang semula berupa kalimat-kalimat yang berisi kata-kata. Proses tokenisasi diawali dengan menghilangkan beberapa *delimiter* yaitu simbol dan tanda baca yang ada pada teks tersebut seperti @, \$, /, &, #, tanda titik (.), koma (,) tanda tanya (?), dan tanda seru (!). Proses pemotongan string berdasarkan tiap kata yang menyusunnya, umumnya setiap kata akan terpisahkan dengan karakter

spasi. Proses tokenisasi mengandalkan karakter spasi pada dokumen teks untuk melakukan pemisahan. Hasil dari proses ini adalah kumpulan kata.

d) *Stemming*

Stemming dilakukan dalam pengolahan data teks untuk mendapatkan kata dasar dari sebuah kata yang telah mengalami imbuhan dengan asumsi bahwa kata-kata tersebut sebenarnya memiliki makna dan arti yang sama. Algoritma ini bekerja berdasarkan morfologi struktural dalam kalimat bahasa Indonesia, yang terdiri atas awalan, akhiran, sisipan, dan awalan+akhiran. Inti dari tahap ini memiliki tujuan meliputi: 1) dalam perkara efisiensi, pada *stemming* dilakukan pengurangan jumlah kata dalam dokumen agar mengurangi kebutuhan dalam ruang penyimpanan dan mempercepat dalam melakukan pencarian. 2) dalam perkara efektivitas, *stemming* dilakukan untuk mengurangi *recall* dengan pengurangan bentuk-bentuk kata ke dalam bentuk dasarnya. Sebagai contoh adalah kata “menduduk-i”, “minum-lah”, “per-banding-an”, dan sebagainya.

e) *Filtering*

Filtering adalah proses mengambil kata-kata penting dari hasil token. Algoritma *stopword* (membuang kata yang kurang penting) atau *wordlist* (menyimpan kata yang penting) dapat digunakan pada tahap ini. *Stopword* adalah kata-kata yang tidak deskriptif dan bukan merupakan kata penting dari suatu dokumen sehingga dapat dibuang (Putranti, N. D. dan Winarko, E., 2014). Contoh *stopword* adalah “yang”, “dan”, “di”, “dari”, dan seterusnya. Dalam filtrasi ini menggunakan *stopword* agar kata-kata yang kurang penting dan sering muncul dalam suatu dokumen dibuang sehingga hanya menyisakan kata-kata yang penting dan mempunyai arti untuk diproses ke tahap selanjutnya.

2.2.6 Pembobotan Kata

Pembobotan kata atau *term weighting* merupakan salah satu tahapan yang perlu diperhatikan dalam mencari informasi dari koleksi dokumen yang heterogen. Dalam dokumen umumnya terdapat kata, frasa, atau unit indeks lainnya yang

menunjukkan konteks dari dokumen tersebut, hal inilah yang disebut *term*. *Term weighting* digunakan untuk memberikan indikator dari setiap kata sesuai dengan tingkat kepentingan masing-masing kata dalam dokumen (Zafikri, A., 2008). Salah satu metode pembobotan kata terbaru yang paling banyak digunakan adalah metode *Term Frequency – Inverse Document Frequency* (TF-IDF). Dalam TF-IDF, perhitungan bobot *term* dari sebuah dokumen dilakukan dengan menghitung masing-masing nilai *Term Frequency* dan *Inverse Document Frequency*.

1. *Term Frequency* (TF)

Term Frequency merupakan faktor yang menentukan perhitungan bobot *term* berdasarkan jumlah dan bentuk kemunculan kata pada dokumen. Pada dasarnya, dapat dikatakan bahwa semakin besar nilai jumlah kemunculan suatu *term*, maka semakin besar juga nilai bobot *term* tersebut dalam dokumen. Menurut Zafikri (2008), perhitungan dalam pembobotan nilai *Term Frequency* dapat dilakukan dengan beberapa cara sebagai berikut.

- a) TF biner, pemberian bobot *term* dilihat berdasarkan ada tidaknya suatu kata dalam dokumen. Jika terdapat kata tersebut maka diberi nilai satu, jika tidak diberi nilai nol.
- b) TF murni (*raw TF*), pemberian bobot *term* dilihat berdasarkan jumlah kemunculan suatu kata dalam dokumen. Misal, jika kata tersebut muncul tiga kali maka diberi bobot tiga.
- c) TF logaritmit, pemberian bobot *term* pada dokumen yang memiliki sedikit kata dalam *query*, namun mempunyai frekuensi yang tinggi.

$$tf = 1 + \log(tf) \quad (2.1)$$

- d) TF normalisasi, pemberian bobot *term* diperoleh dengan membandingkan frekuensi sebuah kata dengan jumlah seluruh kata dalam dokumen.

$$tf = 0,5 + 0,05 \times \left(\frac{tf}{\max tf} \right) \quad (2.2)$$

2. *Inverse Document Frequency* (IDF)

Inverse Document Frequency merupakan proses mengurangi dominasi *term* umum yang sering muncul dalam dokumen. *Term* umum perlu dihilangkan karena sering menyebabkan analisis menjadi kurang maksimal. Selain itu, IDF juga

bertujuan untuk menjaga faktor kejarangmunculan kata (*term scarcity*). Pembobotan dalam IDF dilakukan dengan menghitung nilai faktor kebalikan dari frekuensi dokumen yang mempunyai suatu kata. Adapun perhitungan nilai *Inverse Document Frequency* dapat dilakukan dengan persamaan (2.3) berikut.

$$df_j = \log\left(\frac{D}{df_j}\right) \quad (2.3)$$

di mana

D = jumlah keseluruhan dokumen

df_j = jumlah dokumen yang memiliki *term* t_j

Adapun nilai TF-IDF diperoleh dari perkalian nilai *Term Frequency* dengan nilai *Inverse Document Frequency*. Maka pada perhitungan TF-IDF untuk *raw* TF menggunakan persamaan (2.4) berikut.

$$w_{ij} = tf_{ij} \times df_j \quad (2.4)$$

di mana

w_{ij} = bobot *term* t_j terhadap d_i

tf_{ij} = jumlah kemunculan *term* t_j dalam dokumen d_i

2.2.7 Teorema Bayes

Teorema Bayes merupakan teorema yang mengacu pada konsep probabilitas bersyarat (Tan dkk., 2006). Metode ini merupakan pendekatan statistik untuk melakukan inferensi induksi pada persoalan klasifikasi. Misalkan A dan B adalah kejadian dalam ruang sampel. Larose, D.T. (2005) menyatakan probabilitas bersyarat dalam persamaan (2.5).

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad (2.5)$$

Dimana $P(A \cap B)$ adalah probabilitas interaksi A dan B dan $P(B)$ adalah probabilitas B. Demikian pula $P(B|A) = \frac{P(A \cap B)}{P(A)}$, sehingga nilai $P(A \cap B) = P(B|A) P(A)$. Nilai $P(A \cap B)$ kemudian disubstitusikan ke dalam persamaan (2.5), maka diperoleh persamaan (2.6).

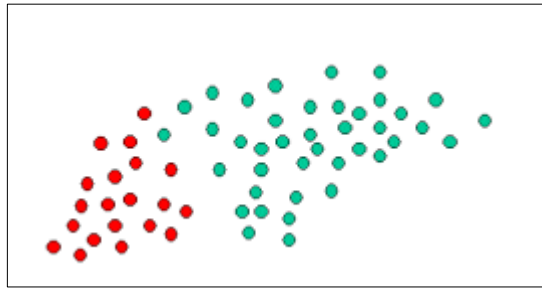
$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \quad (2.6)$$

2.2.8 Naïve Bayes Classifier

Klasifikasi dibagi menjadi lima kelompok berdasarkan teori yang diadopsi atau teori yang menjadi dasar teknik klasifikasi. Lima pengelompokan klasifikasi itu adalah *Bayes's theorem classifier*, *distance-based classifier*, *discriminant classifier*, *neural networks classifier*, dan *decision tree classifier* (Shah, C. dan Jivani, A.G., 2013). Pada penelitian ini akan fokus terhadap *Bayes's theorem classifier* atau algoritma klasifikasi yang mengadopsi teorema Bayes, yaitu *Naïve Bayes Classifier*.

Naïve Bayes Classifier merupakan metode klasifikasi menggunakan model probabilitas sederhana berdasarkan penggunaan teorema Bayes dengan asumsi independen yang kuat (naïve). *Naïve Bayes Classifier* mengasumsikan ada atau tidaknya suatu fitur tertentu pada sebuah kelas, tidak mempengaruhi keberadaan fitur lainnya. Sebagai contoh, sebuah buah disimpulkan sebagai apel jika memiliki warna merah, berbentuk bulat bulat, dan berdiameter 4 cm. Meskipun karakteristik tersebut bergantung satu sama lain, *Naïve Bayes Classifier* memungkinkan semua karakteristik tersebut secara independen berkontribusi dalam probabilitas bahwa buah tersebut adalah buah apel. Atau dengan kata lain, tidak hanya buah apel yang memiliki karakteristik seperti yang disebutkan di atas.

Contoh sederhana dalam perhitungan *Naïve Bayes Classifier*, misalnya terlihat pada Gambar 2.1 terdapat dua kumpulan data yaitu merah dan hijau (Manning, C.D. dkk., 2008). Data baru akan ditambahkan, dan akan ditentukan data baru tersebut merupakan bagian dari kelas yang mana. Karena jumlah data hijau dua kali lebih banyak daripada merah, maka diasumsikan bahwa data yang baru memiliki probabilitas menjadi anggota hijau dua kali lebih besar dari merah. Dalam analisis Bayesian, keyakinan ini dikenal sebagai probabilitas prior. Probabilitas prior didasarkan pada pengalaman sebelumnya, dalam hal ini yakni persentase data merah dan hijau.



Gambar 2.1 Kelompok data merah dan hijau

$$\text{Probabilitas prior untuk data merah} = \frac{\text{jumlah data merah}}{\text{jumlah keseluruhan data}}$$

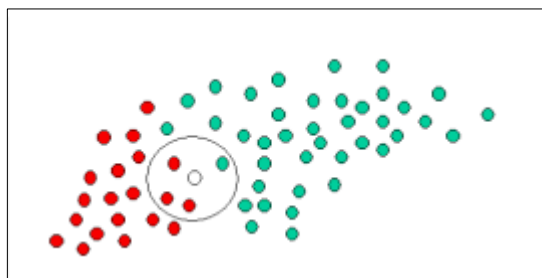
$$\text{Probabilitas prior untuk data hijau} = \frac{\text{jumlah data hijau}}{\text{jumlah keseluruhan data}}$$

Dengan asumsi jumlah keseluruhan data yaitu 60, dengan 20 data merah dan 40 data hijau, maka probabilitas prior untuk keanggotaan kelas yaitu:

$$\text{Probabilitas prior untuk data merah} = \frac{20}{60} = 0,33$$

$$\text{Probabilitas prior untuk data hijau} = \frac{40}{60} = 0,66$$

Setelah menentukan probabilitas prior, data baru (lingkaran putih) akan ditentukan keanggotaanya seperti terlihat pada Gambar 2.2 (Manning, C.D. dkk., 2008). Diasumsikan bahwa semakin banyak suatu kelompok data tertentu (merah atau hijau) di sekitar data baru tersebut, maka kemungkinan data baru mempunyai keanggotaan sesuai kelompok data tersebut semakin besar.



Gambar 2.2 Penambahan data baru pada kelompok data

Untuk mengukur kemungkinan tersebut, digambarkan lingkaran di sekitar data baru X (putih), kemudian jumlah poin dalam lingkaran milik masing-masing label kelas akan dihitung. Dari sini didapatkan:

$$\text{Probabilitas X adalah merah} = \frac{\text{jumlah data merah di sekitar X}}{\text{jumlah data merah}}$$

$$\text{Probabilitas X adalah hijau} = \frac{\text{jumlah data hijau di sekitar X}}{\text{jumlah data hijau}}$$

Sehingga,

$$\text{Probabilitas X adalah merah} = \frac{3}{20} = 0,15$$

$$\text{Probabilitas X adalah hijau} = \frac{1}{40} = 0,02$$

Meskipun probabilitas prior sebelumnya menunjukkan bahwa X merupakan anggota hijau (di mana jumlah hijau dua kali lebih banyak dibandingkan dengan jumlah merah). Kemungkinan setelahnya menunjukkan hal yang sebaliknya; bahwa keanggotaan kelas X adalah merah (di mana terdapat lebih banyak data merah di sekitar X daripada data hijau). Dalam analisis Bayesian, klasifikasi akhir yang dihasilkan ialah penggabungan kedua sumber informasi tersebut.

$$\text{Probabilitas posterior untuk data merah} = \frac{2}{6} \times \frac{3}{20} = \frac{6}{120} = \frac{1}{20} = 0,05$$

$$\text{Probabilitas posterior untuk data hijau} = \frac{4}{6} \times \frac{1}{40} = \frac{4}{240} = \frac{1}{60} = 0,01$$

Sehingga, dapat disimpulkan bahwa data baru tersebut merupakan bagian dari kelompok data merah karena memiliki probabilitas posterior merah yang lebih besar. Berdasarkan sifat dari model probabilitas, *Naïve Bayes Classifier* dapat diterapkan dengan sangat efisien pada *supervised learning setting*. Secara aplikatif, estimasi parameter untuk model *Naïve Bayes Classifier* adalah dengan menggunakan metode *maximum likelihood*, dengan kata lain penggunaan model *Naïve Bayes* dapat dilakukan dengan memahami dulu mengenai aturan Bayesian.

2.2.9 Confusion Matrix

Confusion matrix adalah sebuah tabel yang menyatakan jumlah data uji yang benar diklasifikasikan dan jumlah data uji yang salah diklasifikasikan. Contoh *confusion matrix* untuk klasifikasi biner ditunjukkan pada Tabel 2.2 berikut.

Tabel 2.2 *Confusion matrix* untuk klasifikasi biner

		Kelas Prediksi	
		1	0
Kelas Sebenarnya	1	TP	FN
	0	FP	TN

Keterangan untuk Tabel 2.2 dinyatakan sebagai berikut:

- True Positive* (TP), yaitu jumlah dokumen dari kelas 1 yang benar dan diklasifikasikan sebagai kelas 1.
- True Negative* (TN), yaitu jumlah dokumen dari kelas 0 yang benar diklasifikasikan sebagai kelas 0.
- False Positive* (FP), yaitu jumlah dokumen dari kelas 0 yang salah diklasifikasikan sebagai kelas 1.
- False Negative* (FN) yaitu jumlah dokumen dari kelas 1 yang salah diklasifikasikan sebagai kelas 0.

Perhitungan akurasi dinyatakan dalam persamaan (2.7) berikut.

$$Akurasi = \frac{TP+TN}{TP+FN+FP+TN} \times 100\% \quad (2.7)$$

Kemudian dapat juga menghitung APER (*Apparent Error Rate*) atau disebut juga sebagai laju *error*, yang mana merupakan ukuran evaluasi yang digunakan untuk melihat peluang kesalahan klasifikasi yang dihasilkan oleh suatu fungsi klasifikasi. Semakin kecil nilai APER, maka hasil pengklasifikasian semakin baik (Prasetyo, E., 2013). Formulasi untuk menghitung APER dituliskan dalam persamaan (2.8) berikut.

$$APER = \frac{FP+FN}{TP+FN+FP+TN} \times 100\% \quad (2.8)$$

2.2.10 Asosiasi Teks

Asosiasi teks diperoleh dengan melakukan pendekatan pada perhitungan nilai korelasi. Pada umumnya, nilai korelasi digunakan dalam menyatakan hubungan dua atau lebih variabel kuantitatif. Namun pada asosiasi teks, nilai korelasi dimaknai sebagai keeratan hubungan antar dua atau lebih variabel kualitatif (Ulwan, M.N., 2016). Korelasi bertujuan untuk menemukan tingkat hubungan antara variabel bebas (X) dan variabel bebas (Y), dalam ketentuan data memiliki syarat-syarat tertentu (Abdullah, D. dan Fadlisyah, F., 2014). Perhitungan nilai korelasi pada asosiasi teks menggunakan persamaan (2.9) berikut.

$$r = \frac{n\sum x_i y_i - (\sum x_i) \cdot (\sum y_i)}{\sqrt{\{n\sum x_i^2 - (\sum x_i)^2\} \cdot \{n\sum y_i^2 - (\sum y_i)^2\}}} \quad (2.9)$$

dengan

r = nilai korelasi antara variabel x dan variabel y

n = banyaknya pasangan data x dan y

$\sum x_i$ = jumlah nilai pada variabel x ; $i = 1, 2, 3, \dots, n$

$\sum y_i$ = jumlah nilai pada variabel y

$\sum x_i^2$ = kuadrat dari jumlah nilai pada variabel x

$\sum y_i^2$ = kuadrat dari jumlah nilai pada variabel y

$\sum x_i \cdot \sum y_i$ = total dari hasil perkalian antara nilai variabel x dan variabel y

Dalam perhitungan asosiasi teks, pertama-tama data teks ditransformasikan ke dalam *document-term matrix* (dtm). Adapaun simulasi perhitungan dilakukan pada enam data sebagai berikut.

kata1	kata2	kata3	kata4	kata5
kata1	kata2	kata3	kata4	kata5
kata1	kata2	kata3	kata4	kata5
kata1	kata2	kata3	kata4	kata5
kata1	kata2	kata3	kata4	kata5

Kemudian keenam kata tersebut diubah menjadi *document term matrix* (dtm).

Docs	kata1	kata2	kata3	kata4	kata5
1	1	2	3	4	5
2	2	3	4	5	6
3	3	4	5	6	7
4	4	5	6	7	8
5	5	6	7	8	9

Setelah diperoleh nilai *document term matrix*, selanjutnya dilakukan perhitungan nilai asosiasi. Nilai asosiasi diperoleh dengan menghitung rumus korelasi seperti pada simulasi kata 2 dan kata 4 berikut.

	x_i	y_i	x_i^2	y_i^2	$x_i y_i$
Docs	kata2	kata4	kata2 ²	kata4 ²	kata2×kata4
1	2	4	4	16	8
2	3	5	9	25	15
3	4	6	16	36	24
4	5	7	25	49	35
5	6	8	36	64	48
Total	20	30	90	190	130

$$r = \frac{n \sum x_i y_i - (\sum x_i) \cdot (\sum y_i)}{\sqrt{\{n \sum x_i^2 - (\sum x_i)^2\} \cdot \{n \sum y_i^2 - (\sum y_i)^2\}}}$$

$$r = \frac{(6 \times 130) - (20 \times 30)}{\sqrt{\{(6 \times 90) - 20^2\} \cdot \{(6 \times 190) - 30^2\}}}$$

$$r = \frac{180}{\sqrt{140 \cdot 240}} = \frac{180}{\sqrt{33.600}} = \frac{180}{183,3} = 0,98$$

Jadi, nilai korelasi kata 2 dan kata 4 sebesar 0,98. Hal ini menunjukkan bahwa besarnya asosiasi atau hubungan antara kata 2 dan kata 4 sebesar 98%.

BAB III

METODE KERJA PRAKTIK

3.1 Gambaran Umum Instansi

Perusahaan Listrik Negara (disingkat PLN) atau nama resminya adalah PT PLN (Persero) adalah sebuah BUMN yang mengurus semua aspek kelistrikan yang ada di Indonesia. Berawal di akhir abad 19, bidang pabrik gula dan pabrik ketenagalistrikan di Indonesia mulai ditingkatkan saat beberapa perusahaan asal Belanda yang bergerak di bidang pabrik gula dan pabrik teh mendirikan pembangkit tenaga listrik untuk keperluan sendiri. Antara tahun 1942-1945 terjadi peralihan pengelolaan perusahaan-perusahaan Belanda tersebut oleh Jepang, setelah Belanda menyerah kepada pasukan tentara Jepang di awal Perang Dunia II.

Proses peralihan kekuasaan kembali terjadi di akhir Perang Dunia II pada Agustus 1945, saat Jepang menyerah kepada Sekutu. Kesempatan ini dimanfaatkan oleh para pemuda dan buruh listrik melalui delegasi Buruh/Pegawai Listrik dan Gas yang bersama-sama dengan Pemimpin KNI Pusat berinisiatif menghadap Presiden Soekarno untuk menyerahkan perusahaan-perusahaan tersebut kepada Pemerintah Republik Indonesia. Pada 27 Oktober 1945, Presiden Soekarno membentuk Jawatan Listrik dan Gas di bawah Departemen Pekerjaan Umum dan Tenaga dengan kapasitas pembangkit tenaga listrik sebesar 157,5 MW.

Pada tanggal 1 Januari 1961, Jawatan Listrik dan Gas diubah menjadi BPU-PLN (Badan Pemimpin Umum Perusahaan Listrik Negara) yang bergerak di bidang listrik, gas dan kokas yang dibubarkan pada tanggal 1 Januari 1965. Pada saat yang sama, 2 (dua) perusahaan negara yaitu Perusahaan Listrik Negara (PLN) sebagai pengelola tenaga listrik milik negara dan Perusahaan Gas Negara (PGN) sebagai pengelola gas diresmikan. Pada tahun 1972, sesuai dengan Peraturan Pemerintah No. 17, status Perusahaan Listrik Negara (PLN) ditetapkan sebagai Perusahaan Umum Listrik Negara dan sebagai Pemegang Kuasa Usaha Ketenagalistrikan (PKUK) dengan tugas menyediakan tenaga listrik bagi kepentingan umum.

Seiring dengan kebijakan Pemerintah yang memberikan kesempatan kepada sektor swasta untuk bergerak dalam bisnis penyediaan listrik, maka sejak tahun 1994 status PLN beralih dari Perusahaan Umum menjadi Perusahaan Perseroan (Persero) dan juga sebagai PKUK dalam menyediakan listrik bagi kepentingan umum hingga sekarang. Adapun alamat kantor pusat PT PLN (Persero) berada di Jalan Trunojoyo Blok M – I No 135 Kebayoran Baru, Jakarta 12160, Indonesia seperti terlihat pada Gambar 3.1 berikut.



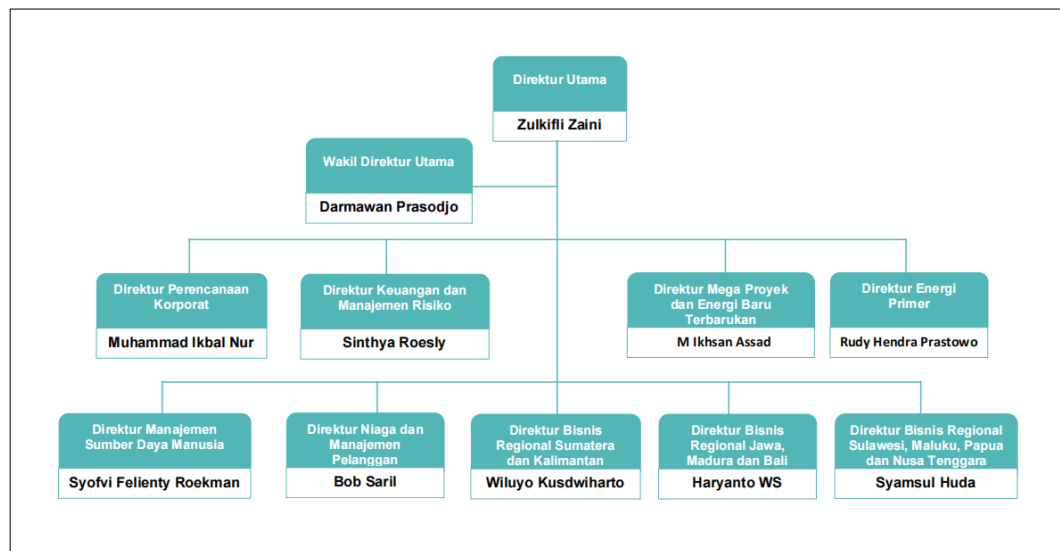
Gambar 3.1 Denah kantor pusat PT PLN (Persero)

3.1.1 Visi, Misi, dan Struktur Organisasi

Visi PT PLN (Persero) adalah menjadi perusahaan listrik terkemuka se-Asia Tenggara dan nomor 1 pilihan pelanggan untuk solusi energi. Sedangkan untuk misi PT PLN (Persero) antara lain:

- a) Menjalankan bisnis kelistrikan dan bidang lain yang terkait, berorientasi pada kepuasan pelanggan, anggota perusahaan dan pemegang saham.
- b) Menjadikan tenaga listrik sebagai media untuk meningkatkan kualitas kehidupan masyarakat.
- c) Mengupayakan agar tenaga listrik menjadi pendorong kegiatan ekonomi.
- d) Menjalankan kegiatan usaha yang berwawasan lingkungan.

Kantor pusat PT PLN (Persero) menggunakan bentuk struktur organisasi lini. Manajer sebagai pimpinan tertinggi dan pemegang kendali penuh terhadap kinerja kantor pusat PT PLN (Persero). Secara sistematis, struktur organisasi lini yang ada pada PT PLN (Persero) dapat dilihat pada Gambar 3.2 berikut.



Gambar 3.2 Struktur organisasi kantor pusat PT PLN (Persero)

Dalam melaksanakan kegiatan bisnis PT PLN (Persero), tiap lini divisi memiliki tugas dan tanggung jawabnya masing-masing. Adapun tugas dan tanggung jawab masing-masing lini tersebut dapat diuraikan sebagai berikut:

a) Direktur Utama

Direktur Utama bertanggung jawab memimpin, membina dan mengelola PLN sesuai dengan maksud dan tujuan PLN dan senantiasa meningkatkan kinerja PLN untuk mencapai visi serta menguasai, memelihara, dan mengelola kekayaan PLN.

b) Wakil Direktur Utama

Wakil Direktur Utama bertanggung jawab memimpin dan membina fungsi manajemen perubahan, manajemen digital, manajemen regulator, fungsi perencanaan dan pengembangan listrik pedesaan serta memastikan PLN mampu berkembang menuju terwujudnya visi, misi, dan tujuan PLN.

c) Direktur Perencanaan Korporat

Direktur Perencanaan Korporat bertanggung jawab memimpin, membina, dan mengelola Direktorat Perencanaan Korporat yang meliputi fungsi perencanaan sistem, perencanaan strategi korporat, manajemen portofolio, kepatuhan, dan pengendalian kinerja korporat untuk memastikan PLN mampu berkembang menuju terwujudnya visi, misi, dan tujuan PLN.

d) Direktur Keuangan dan Manajemen Risiko

Direktur Keuangan dan Manajemen Risiko bertanggung jawab memimpin, membina, dan mengelola Direktorat Keuangan dan Manajemen Risiko yang meliputi fungsi keuangan, anggaran, perbendaharaan, akuntansi, sistem dan teknologi informasi, manajemen risiko, kebijakan korporat, serta manajemen aset untuk memastikan PLN mampu berkembang menuju terwujudnya visi, misi, dan tujuan PLN.

e) Direktur Mega Proyek dan Energi Baru Terbarukan

Direktur Mega Proyek dan Energi Baru Terbarukan bertanggung jawab memimpin, membina, dan mengelola Direktorat Mega Proyek dan Energi Baru Terbarukan yang meliputi fungsi perencanaan dan *engineering* konstruksi, perencanaan dan *engineering* energi baru dan terbarukan (EBT), pengadaan *Engineering, Procurement, and Construction* (EPC) dan *Independent Power Producer* (IPP) energi baru dan terbarukan (EBT), konstruksi pembangkit, transmisi dan gardu induk, *project management office*, dan manajemen rantai pasok untuk memastikan PLN mampu berkembang menuju terwujudnya visi, misi, dan tujuan PLN.

f) Direktur Energi Primer

Direktur Energi Primer bertanggung jawab memimpin, membina, dan mengelola Direktorat Energi Primer yang meliputi fungsi pengadaan batubara, gas dan bahan bakar minyak (BBM), dan *Independent Power Producer* (IPP) untuk memastikan PLN mampu berkembang menuju terwujudnya visi, misi, dan tujuan PLN.

g) Direktur Manajemen Sumber Daya Manusia

Direktur Manajemen Sumber Daya Manusia bertanggung jawab memimpin, membina, dan mengelola Direktorat Manajemen Sumber Daya Manusia yang meliputi fungsi strategi *human capital*, pengembangan talenta, pelayanan *human capital*, umum, keselamatan dan kesehatan kerja (K3), keamanan dan lingkungan, komunikasi korporat dan *Corporate Social Responsibility* (CSR), dan hukum korporat serta pembinaan dan tanggung jawab kinerja Pusat Pendidikan dan Pelatihan (*Corporate University*) untuk memastikan PLN mampu berkembang menuju terwujudnya visi, misi, dan tujuan PLN.

h) Direktur Niaga dan Manajemen Persaingan

Direktur Niaga dan Manajemen Persaingan bertanggung jawab memimpin, membina, dan mengelola Direktorat Keuangan dan Manajemen Risiko yang meliputi fungsi pusat keunggulan niaga, pemasaran dan pengembangan produk, manajemen persaingan korporat dan institusi besar, pelayanan pelanggan retail, serta tarif dan subsidi untuk memastikan PLN mampu berkembang menuju terwujudnya visi, misi, dan tujuan PLN.

i) Direktur Bisnis Regional Sumatera dan Kalimantan

Direktur Bisnis Regional Sumatera dan Kalimantan bertanggung jawab memimpin, membina, dan mengelola Direktorat Bisnis Regional Sumatera dan Kalimantan yang meliputi perencanaan dan pengendalian strategi regional, pembangkitan dan energi baru & terbarukan (EBT), transmisi, distribusi, dan retail dengan wilayah Pulau Sumatera dan Kalimantan untuk memastikan PLN mampu berkembang menuju terwujudnya visi, misi, dan tujuan PLN.

j) Direktur Bisnis Regional Jawa, Madura, dan Bali

Direktur Bisnis Regional Jawa, Madura, dan Bali bertanggung jawab memimpin, membina, dan mengelola Direktorat Bisnis Regional Sumatera dan Kalimantan yang meliputi perencanaan dan pengendalian strategi regional, pembangkitan dan energi baru & terbarukan (EBT), transmisi, distribusi, dan retail dengan wilayah Pulau Jawa, Madura, dan Bali untuk

memastikan PLN mampu berkembang menuju terwujudnya visi, misi, dan tujuan PLN.

- k) Direktur Bisnis Regional Sulawesi, Maluku, Papua, dan Nusa Tenggara
Direktur Bisnis Regional Sulawesi, Maluku, Papua, dan Nusa Tenggara bertanggung jawab memimpin, membina, dan mengelola Direktorat Bisnis Regional Sumatera dan Kalimantan yang meliputi perencanaan dan pengendalian strategi regional, pembangkitan dan energi baru & terbarukan (EBT), transmisi, distribusi, dan retail dengan wilayah Pulau Sulawesi, Maluku, Papua, dan Nusa Tenggara untuk memastikan PLN mampu berkembang menuju terwujudnya visi, misi, dan tujuan PLN.

3.1.2 Layanan dan Produk

PLN adalah perusahaan BUMN yang menjadi pemasok kebutuhan listrik bagi seluruh masyarakat Indonesia. Pembayaran produk-produk PLN saat ini semakin mudah karena dapat dilakukan dengan *online*. Saat ini, PLN telah memiliki dua produk unggulan, yaitu listrik Prabayar dan listrik Pascabayar. Pada tahun 2010, PLN mengeluarkan kebijakan baru untuk pembayaran listrik, dari pembayaran listrik dengan pascabayar yaitu pelanggan menggunakan energi listrik dulu dan membayar pada bulan berikutnya. Setiap bulan PLN harus mencatat meter, menghitung, dan menerbitkan rekening yang harus dibayar oleh pelanggan, serta melakukan penagihan kepada pelanggan yang terlambat atau tidak membayar, dan memutus aliran listrik jika konsumen terlambat atau tidak membayar rekening listrik setelah waktu tertentu.

Sekarang, pada sistem listrik pintar (Prabayar), pelanggan mengeluarkan uang terlebih dulu untuk membeli energi listrik yang akan digunakannya. Besaran energi listrik yang telah dibeli oleh pelanggan dimasukkan ke dalam Meter Prabayar (MPB) yang terpasang di lokasi pelanggan melalui sistem token atau pulsa. Penggantian yang dilakukan jika berganti ke layanan listrik Prabayar hanya mengganti kWh meter yang dahulu analog, hanya untuk menghitung besarnya energi listrik yang terpakai, sedangkan kWh meter listrik Prabayar menggunakan kWh khusus yang bisa dimasukan token listrik dan ketika token listrik habis maka

listrik akan otomatis terputus. Namun PLN tidak mewajibkan pelanggan menggunakan listrik Prabayar, PLN hanya memberikan pilihan kepada pelanggan untuk menggunakan listrik Prabayar atau Pascabayar.

Penghitungan kWh Meter Listrik Pintar sama saja dengan kWh Meter Analog karena telah melalui tahap standarisasi Tera (tidak lebih mahal) dan harga Rupiah/kWh Listrik sudah diatur dalam penyesuaian tarif tenaga listrik melalui Peraturan Menteri ESDM Nomor 31 Tahun 2014 Tentang Tarif Tenaga Listrik yang Disediakan oleh PT PLN (Persero). Kemudian yang akan menentukan hemat atau boros adalah perilaku penggunaan peralatan listrik oleh pelanggan. Serupa dengan telepon, sistem Prabayar dapat membuat pelanggan berhemat. Sebaliknya dengan Pascabayar, pelanggan cenderung lebih boros karena kurang terkendali. Produk yang dijual oleh PLN adalah energi yang bisa dipergunakan untuk menjalankan mesin dan penerangan. Berdasarkan tujuan pemakaian listrik, maka produk energi listrik yang dijual kepada pelanggan dibagi ke dalam beberapa golongan tarif S (kepentingan sosial), tarif R (rumah tangga kecil dan besar), tarif B (bisnis), tarif I (industri), tarif P (pemerintah dan penerangan jalan umum), tarif T (traksi), dan tarif C (curah).

a) Pelanggan Sosial

Yang termasuk dalam golongan tarif S (sosial) yaitu pelanggan badan sosial yang tenaga listriknya digunakan untuk kepentingan atau kegiatan sosial. Tarif tenaga listrik untuk keperluan pelayanan sosial terdiri atas:

- S-1/TR (Tegangan Rendah), 0 s/d 449 VA.
- S-2/TR (Tegangan Rendah), 450 VA s/d 200 kVA.
- S-3/TM (Tegangan Menengah), di atas 200 kVA.

Khusus pelanggan golongan tarif S-3 dibedakan menjadi kegiatan sosial murni dan kegiatan sosial komersial sebagai berikut.

1. Kegiatan Sosial Murni

- Rumah sakit milik instansi pemerintah pusat atau daerah.
- Tempat ibadah (masjid, gereja, vihara, kelenteng, atau sejenisnya).
- Panti sosial (yatim-piatu, jompo)
- Pusat rehabilitasi sosial (narkotika, penyakit kusta, atau sejenisnya).

- Asrama pelajar atau mahasiswa milik pemerintah.
- Asrama haji pemerintah.
- Pusat pendidikan keagamaan (pondok pesantren atau sejenisnya).
- Gedung kantor partai politik dan afiliasi.
- Museum milik pemerintah pusat atau daerah.
- Kebun binatang milik pemerintah pusat atau daerah.

2. Kegiatan Sosial Komersial

- Sekolah atau perguruan tinggi swasta.
- Rumah sakit swasta.
- Poliklinik atau tempat praktik dokter bersama.
- Lembaga riset swasta.
- Yayasan pengelola haji non-pemerintah.
- Pusat pendidikan dan latihan perusahaan swasta. Misalnya: pusdiklat Garuda, pusdiklat Bank Mandiri, Pusdiklat Unilever, Lembaga Pendidikan Indonesia – Amerika, dll.

b) Pelanggan Rumah Tangga

Yang merupakan pelanggan rumah tangga adalah pelanggan perseorangan atau sosial yang tenaga listriknya digunakan untuk keperluan rumah tangga seperti:

- Rumah untuk tempat tinggal.
- Kelompok rumah kontrakan.
- Rumah susun milik perorangan.
- Rumah susun milik perumnas.
- Asrama keluarga pegawai perusahaan swasta.
- Asrama mahasiswa.

c) Pelanggan Bisnis

Yang termasuk dalam pelanggan bisnis adalah pelanggan yang sebagian atau seluruh tenaga listrik dari PT PLN (Persero) digunakan untuk kegiatan dalam bentuk-bentuk berikut:

- Usaha jual beli barang, jasa, dan perhotelan.

- Usaha perbankan.
- Usaha perdagangan ekspor dan impor.
- Kantor Firma, CV, PT, atau badan hukum/perorangan yang bergerak dalam bidang usaha perdagangan.
- Usaha pergudangan di mana sebagian atau seluruh bangunan digunakan untuk tempat penyimpanan barang atau material.
- Usaha perorangan atau badan hukum yang sebagian besar atau seluruh kegiatannya merupakan penjualan barang atau jasa.
- Usaha-usaha lainnya yang bertendensi komersial seperti praktik dokter, dan lain sebagainya.

Beda dengan aturan tarif pelanggan sebelumnya, usaha dengan kegiatan pengolahan yang memberikan nilai tambah atas sesuatu produk, bisa dikeluarkan dari tarif kelompok bisnis dan dimasukkan kedalam tarif industri. Sesuai dengan kebijakan yang diambil demi konsistensi penerapan Klasifikasi Lapangan Usaha Indonesia (KLUI). Contohnya seperti perbengkelan las, bengkel karoseri, pertukangan dan kerajinan mebel, dan sebagainya.

d) Pelanggan Industri

Yang termasuk dalam pelanggan industri adalah pelanggan yang menggunakan listrik dari PT PLN (Persero) untuk keperluan industri. Beberapa industri yang menjadi pelanggan dari PT PLN (Persero) antara lain:

- PT Gunung Garuda;
- PT Multi Starda Araha Sarana;
- PT Toyogiri;
- PT Mulia Keramik;
- PT Maxxis International;
- dan lain-lain.

Tarif tenaga listrik untuk keperluan industri, terdiri atas:

- I-1/TR (Tegangan Rendah), 450 VA s/d 14 kVA.

- I-2/TR (Tegangan Rendah), di atas 14 kVA s/d 200 kVA.
- I-3/TM (Tegangan Menengah), di atas 200 kVA.
- I-4/TT (Tegangan Tinggi), 30.000 kVA ke atas.

e) Pelanggan Pemerintah

Yang termasuk dalam pelanggan pemerintah adalah pelanggan yang menggunakan listrik untuk keperluan publik seperti:

- Pelayanan publik.
- Kantor pemerintah.
- Penerangan jalan umum.

Tarif tenaga listrik untuk keperluan pemerintah terdiri atas:

- P-1/TR (Tegangan Rendah), 450 VA s/d 5.500 VA.
- P-1/TR (Tegangan Rendah), 6.600 VA s/d 200 kVA.
- P-2/TM (Tegangan Menengah), di atas 200 kVA.
- P-3/TR (Tegangan Rendah), untuk penerangan jalan umum.

f) Pelanggan Traksi

Yang termasuk dalam pelanggan traksi adalah perusahaan yang bergerak dalam bidang transportasi umum yang dijalankan oleh PT Kereta Api Indonesia (KAI), dengan ketentuan sebagai berikut:

- Tenaga listrik dimanfaatkan secara langsung maupun tidak langsung sebagai penggerak utama sarana pengangkutan yang dioperasikan.
- Instalasi untuk operasi transportasi dipisahkan dari instalasi penunjang seperti bangunan gedung stasiun, bengkel pemeliharaan, gudang perlengkapan, dan lain sebagainya.
- Tarif tenaga listrik untuk keperluan traksi adalah T/TM (Tegangan Menengah), di atas 200 kVA.

g) Pelanggan Curah

Yang termasuk dalam pelanggan curah adalah badan usaha KUD (Koperasi Unit Desa) yang memiliki ketentuan sebagai berikut:

- Bergerak di bidang khusus tenaga listrik.

- Mengoperasikan sendiri jaringan tegangan menengah dan tegangan rendah yang memenuhi standar PLN setempat.
- Memegang Izin Usaha Ketenagalistrikan untuk Kepentingan Umum (IUKU) yang sah.
- Bersedia disamakan dengan pelanggan dan tidak menuntut hak eksklusif tertentu.
- Tarif tenaga listrik untuk keperluan curah adalah C/TM (Tegangan Menengah), di atas 200 kVA.
- Tarif tenaga listrik untuk keperluan layanan khusus pada tegangan rendah (TR), tegangan menengah (TM), tegangan tinggi (TT), diperuntuk hanya untuk pelanggan yang memerlukan pelayanan dengan kualitas khusus dan karena berbagai hal (L/TR, TM, TT).

3.1.3 Aturan Bisnis

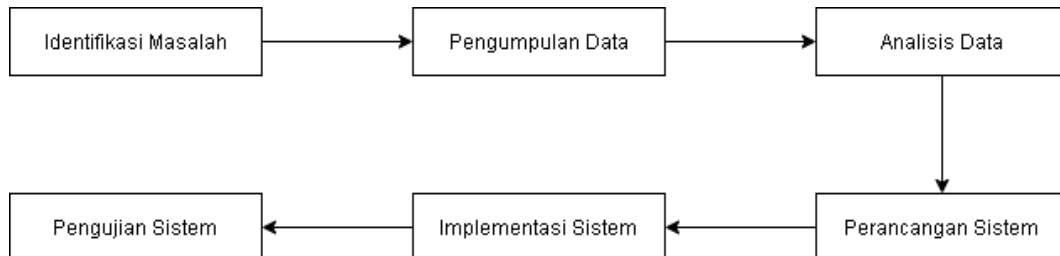
PT PLN (Persero) sebagai Badan Usaha Milik Negara (BUMN) di bidang kelistrikan yang melayani masyarakat di seluruh Nusantara, bertekad untuk memberikan pelayanan jasa ketenagalistrikan yang terbaik dan memenuhi standar ketenagalistrikan yang dapat diterima dunia internasional dan mewujudkan hal itu dengan bertumpu pada kapasitas seluruh warganya. Dalam menjalankan bisnisnya, PLN bertekad bekerja dengan semangat untuk selalu menghasilkan produk dan pelayanan yang terbaik serta memperlakukan pelanggan, mitra usaha, dan pemasok dengan adil tanpa membedakan-bedakannya.

Dalam rangka menjaga agar tetap ada konsistensi dalam penyelenggaraan perusahaan yang baik (*Good Corporate Governance*), manajemen PLN bertekad untuk menumbuh kembangkan kebiasaan dan tata pergaulan profesional yang baik dan sekaligus mencerminkan jati diri PLN yang dapat kita banggakan bersama. Usaha ini juga merupakan perwujudan dari kesungguhan hati warga PLN untuk bekerja dan berusaha selaras dengan falsafah, visi, misi, dan tata nilai perusahaan yang sudah disepakati bersama. Semua ini akan dijalankan dengan tetap mengacu pada aspirasi untuk menciptakan nilai yang maksimal bagi bangsa dan negara Indonesia.

Manajemen PLN juga bertekad untuk menyelenggarakan perusahaan dengan mengajak seluruh anggota PLN dan semua pihak yang peduli dengan kemajuan perusahaan ini, dapat menjaga perusahaan ini agar tetap berkiprah secara bertanggung jawab. Keterbukaan dan partisipasi ini akan dijalankan dengan prinsip bahwa informasi perusahaan dapat diakses dan diperoleh dengan mudah oleh masyarakat dan semua pihak yang berhak, tanpa mengabaikan prinsip kerahasiaan informasi tersebut. Sebaliknya, manajemen perusahaan juga senantiasa membuka diri bagi semua masukan dan saran dari lingkungan internal dan eksternal perusahaan.

3.2 Tahapan Penyelesaian Masalah

Tahapan penyelesaian masalah didefinisikan sebagai pedoman penulis dalam melaksanakan penelitian. Hal ini bertujuan agar hasil yang dicapai tidak menyimpang dan sesuai dengan tujuan yang telah ditetapkan. Adapun diagram alir penelitian dapat dilihat pada Gambar 3.3 berikut.



Gambar 3.3 Diagram alir penelitian

Berdasarkan Gambar 3.3, dapat dijelaskan deskripsi atau uraian dari setiap tahap penelitian sebagai berikut.

3.2.1 Identifikasi Masalah

Pada tahap ini, penulis melakukan persiapan seperti menentukan judul penelitian, rumusan masalah, batasan masalah, tujuan, serta studi pustaka untuk mencari teori-teori yang dapat membantu dalam penyelesaian masalah yang diteliti.

3.2.2 Pengumpulan Data

Pada tahap pengumpulan data, komponen yang diperlukan yaitu data ulasan pengguna aplikasi PLN Mobile. Data ulasan diperoleh dari situs Google Play Store menggunakan teknik *web scraping* dengan format *file* CSV. Setelah memperoleh data, selanjutnya data tersebut akan diolah dalam tahap analisis data.

3.2.3 Analisis Data

Tahap analisis data yaitu proses menerjemahkan tujuan ke dalam spesifikasi kebutuhan sistem. Analisis dilakukan agar dapat mengetahui permasalahan yang muncul pada sistem yang sedang berjalan. Sehingga dapat dibangun sistem yang lebih baik dengan menerapkan solusi dari permasalahan yang terjadi.

3.2.4 Perancangan Sistem

Pada tahap ini, dilakukan perancangan proses atau alur sistem dan perancangan sistem yang akan dibangun. Tahap ini bertujuan untuk menjelaskan proses-proses yang akan dilakukan dan urutan proses yang akan dilakukan. Perancangan sistem meliputi blok diagram alur sistem dan *flowchart* sistem yang akan dibangun.

3.2.5 Implementasi Sistem

Pada tahap implementasi sistem, dilakukan penerapan terhadap rancangan-rancangan yang telah dibuat pada tahap sebelumnya, sehingga dihasilkan suatu luaran berupa sistem yang dapat melakukan klasifikasi data ulasan pengguna aplikasi PLN Mobile. Sistem ini diimplementasikan menggunakan bahasa pemrograman R dan IDE (*Integrated Development Environment*) RStudio.

3.2.6 Pengujian Sistem

Pengujian sistem dilakukan untuk menjelaskan algoritma *Naïve Bayes Classifier* yang mana menghitung probabilitas data dari hasil klasifikasi data, sehingga menghasilkan luaran berupa klasifikasi yang akurat. Pengujian dilakukan untuk mengetahui tingkat akurasi dengan menggunakan *confusion matrix*, sebuah

matrik dari prediksi yang akan dibandingkan dengan kelas yang asli dari data input. *Confusion matrix* menghasilkan data angka dari hasil pengujian sistem. Hasil pengujian digunakan untuk melihat nilai akurasi dari sistem yang dibangun.

3.3 Bahan dan Data

3.3.1 Data yang Diperoleh

Data ulasan pengguna aplikasi PLN Mobile yang digunakan pada penelitian ini berupa data sekunder yang diperoleh dari situs Google Play Store. *Dataset* ini terdiri atas lima kategori meliputi: bintang 1 (sangat buruk), bintang 2 (buruk), bintang 3 (netral), bintang 4 (baik), dan bintang 5 (sangat baik). Total ulasan yang sebenarnya ialah lebih dari 50.000 data, namun pada penelitian ini hanya menggunakan 3.200 data dengan tujuan efisiensi sistem. Selain itu, dataset yang digunakan juga tidak menyertakan kategori bintang 3 (netral) agar sistem dapat menghindari bias pada saat melakukan pembobotan kata. Contoh data ulasan aplikasi PLN Mobile di situs Google Play Store terlihat pada Gambar 3.4 berikut.



Gambar 3.4 Data ulasan aplikasi PLN Mobile

3.3.2 Prosedur Pengumpulan Data

Dataset yang digunakan pada penelitian ini adalah data ulasan aplikasi PLN Mobile yang diambil dari situs Google Play Store dengan alamat URL <https://play.google.com/store/apps/details?id=com.icon.pln123>. *Dataset* diperoleh menggunakan teknik *web scraping* dengan bantuan *library* Google Play Scraper dan *cloud environment* Google Colaboratory. Contoh hasil (atribut data) *web scraping* yang diperoleh berupa ulasan pengguna dapat dilihat pada Tabel 3.1.

Tabel 3.1 Contoh hasil *web scraping*

No.	Ulasan
1.	Bagus sih ada aplikasinya tapi aplikasinya saja bukan pengaplikasiannya. Saat melakukan pengaduan tidak ada regu/teknisi dari pihak PLN yang datang, mau gak mau harus tetap telfon dengan menggunakan pulsa, kalau gak ditelfon gak bakal ada yang datang petugasnya. Mungkin bukan saya sendiri yang merasakan hal seperti ini tapi dimohon lebih dipantau lagi kedepannya.
2.	Saya kasih bintang 1,,,,, Karena token yang saya beli tgl 20 gagal di masukan Sudah hubungi pln wilayah saya di kasih kode token 4 baris di masukan disuruh beli token baru sudah saya laksanakan bisa masuk,, Tapi token yang sebelumnya saya masukan gagal tetep gagal tidak di masukan,,,tidak ada tanggung jawab dari pln tidak dapat ganti rugi 200ribu di kampung kok di suruh mengiklaskan ,,tetep saja saya tidak ikhlas Sangat sangat merugikan pelanggan
3.	Pake aplikasi nya sama saja dn meng hubungi langsung 123, keluhan saya dari tadi mlam jam 8 malam sampe skarang jam 9 pagi blm ada penindakan dari PLN,, keluhan lampu padam satu bangunan ada 11 meteran,, di lokasi kami tinggl kami yang lampunya padam,,di tetangga/birman" lampunya dari jam 8 malam nyala sampe skarang,,,

BAB IV

ANALISIS DAN DESAIN SISTEM

4.1 Analisis Sistem

Dalam membangun sebuah sistem, diperlukan adanya analisis sistem. Analisis sistem dilakukan bertujuan untuk mengidentifikasi kebutuhan-kebutuhan pada sistem, kebutuhan ini meliputi kebutuhan fungsional maupun kebutuhan nonfungsional. Berikut adalah uraian analisis sistem dalam membangun sistem klasifikasi sentimen pada ulasan pengguna aplikasi PLN Mobile.

4.1.1 Analisis Kebutuhan Fungsional

Analisis sistem digunakan untuk menganalisis dan mengidentifikasi permasalahan untuk menghasilkan sistem sesuai dengan yang diinginkan. Sistem yang dibangun akan digunakan untuk mengklasifikasi sentimen pada ulasan pengguna aplikasi PLN Mobile menggunakan algoritma *Naïve Bayes Classifier*. Data ulasan yang akan diproses diperoleh dari situs Google Play Store karena situs tersebut menyediakan data khusus untuk ulasan-ulasan dari berbagai aplikasi Android. Klasifikasi sentimen dilakukan berdasarkan nilai bobot kata positif dan negatif terhadap data ulasan yang telah dibersihkan melalui proses *preprocessing*.

4.1.2 Analisis Kebutuhan Nonfungsional

Analisis kebutuhan sistem secara nonfungsional adalah analisis mengenai kebutuhan pendukung sistem yang akan dibuat, bertujuan untuk memenuhi kebutuhan nonfungsional pada sistem. Kebutuhan secara nonfungsional tersebut meliputi kebutuhan *hardware* (perangkat keras) dan *software* (perangkat lunak) yang digunakan dalam penelitian sebagai berikut.

1. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan untuk merancang dan mengimplementasikan sistem yang dibuat yaitu laptop dengan spesifikasi:

- a) ASUS X450CA
- b) CPU Intel® 1007U 1,5 GHz
- c) GPU Intel® HD Graphics 4000 1536 MB
- d) RAM 2 GB
- e) HDD 500 GB

2. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan untuk merancang dan mengimplementasikan sistem yang dibuat meliputi:

- a) RStudio

RStudio adalah *Integrated Development Environment* (IDE) khusus untuk bahasa pemrograman R yang mana berfungsi dalam menuliskan *script* analisis sentimen pada ulasan pengguna aplikasi PLN Mobile.

- b) Bahasa pemrograman R versi 4.0.5 64-bit

Bahasa pemrograman R digunakan sebagai basis bahasa pemrograman untuk membangun sistem klasifikasi sentimen pada ulasan pengguna aplikasi PLN Mobile. Bahasa R dipilih karena mendukung komputasi statistik dan memiliki banyak *library* yang dibutuhkan dalam membangun sistem klasifikasi sentimen pada ulasan pengguna aplikasi PLN Mobile.

- c) Windows 10 Education versi 10.0 64-bit

Windows 10 Education adalah sistem operasi yang diinstal dalam laptop dan digunakan pada penelitian ini.

4.2 Desain Sistem

Pada tahap ini, penulis melakukan desain atau perancangan sistem meliputi blok diagram alur, *flowchart*, dan *wireframe* sistem. Tujuan dari perancangan ini yaitu dapat memberikan gambaran kepada pengguna tentang sistem yang diusulkan dan memberikan ilustrasi dalam pembuatan sistem nantinya.

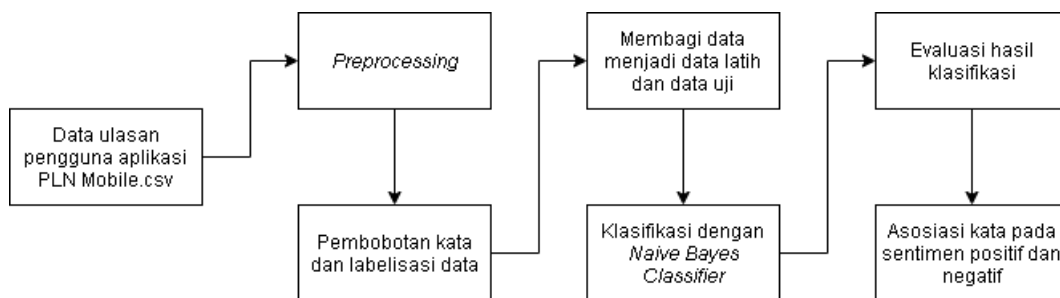
4.2.1 Desain Logik

Perancangan sistem dapat diartikan sebagai gambaran atau sketsa dari alur proses sistem pengolahan *dataset*. Dalam perancangan suatu sistem terdapat proses

aliran sistem dan *flowchart* sistem. Perancangan sistem ini dilakukan sebelum proses implementasi sistem. Berikut adalah rancangan blok diagram alur dan *flowchart* yang dibuat sesuai dengan sistem yang akan dibangun.

a) Blok Diagram Alur

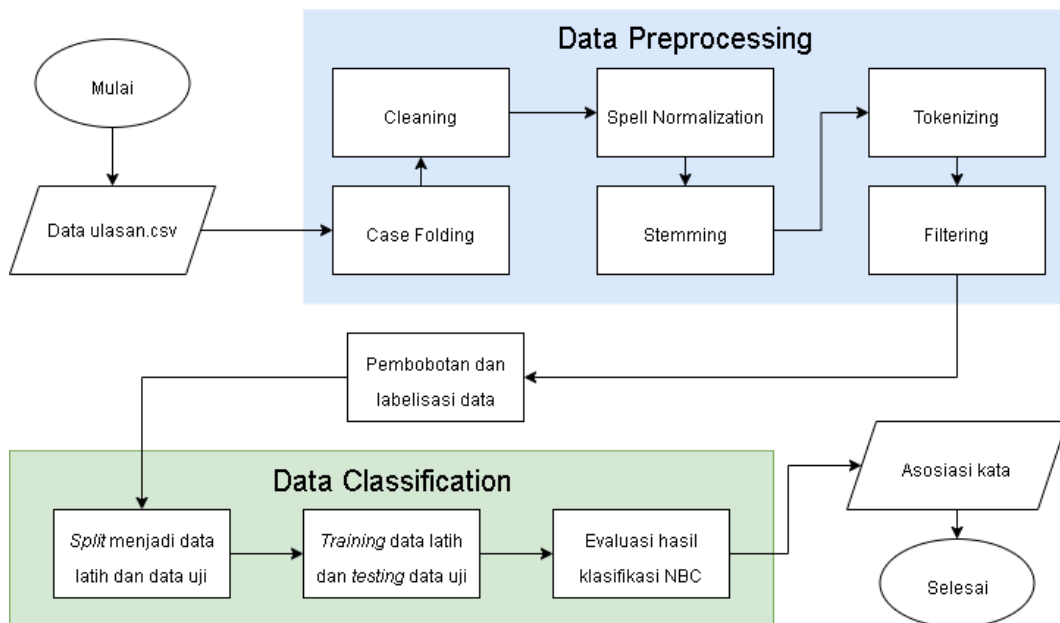
Perancangan sistem berguna untuk menjelaskan bagaimana alur sistem yang akan dibangun. Perancangan blok diagram alur sistem dapat dilihat pada Gambar 4.1.



Gambar 4.1 Blok diagram alur sistem

b) *Flowchart*

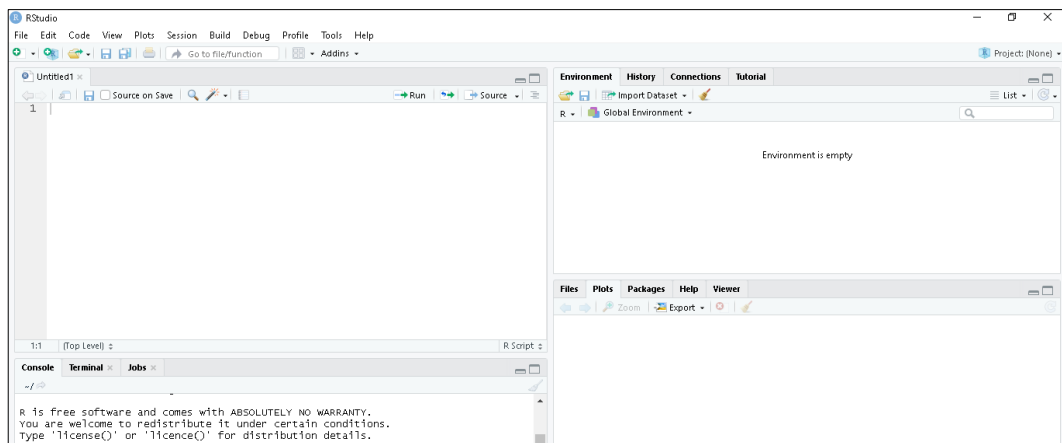
Flowchart akan menjabarkan algoritma yang digunakan pada pembuatan sistem pengklasifikasi sentimen pada ulasan pengguna aplikasi PLN Mobile menggunakan *Naïve Bayes Classifier*. Algoritma dimulai dari memasukkan data ulasan pengguna aplikasi PLN Mobile dengan format *file* CSV, kemudian sistem mulai melakukan *preprocessing* untuk membersihkan dan menormalisasi data. Hasil dari proses *processing* akan dilanjutkan ke proses labelisasi data. Dalam tahap ini, data ulasan diberi nilai bobot pada tiap kata berdasarkan sentimen positif dan negatif. Lalu dari nilai bobot tersebut akan ditentukan apakah sebuah ulasan masuk ke sentimen positif atau sentimen negatif. Setelah itu, data hasil labelisasi akan dibagi menjadi data latih dan data uji. Data latih menjadi model atau referensi untuk mengklasifikasikan sentimen ulasan menjadi positif atau negatif. Kemudian dilakukan pengujian terhadap data uji untuk mengetahui seberapa efektif *Naïve Bayes Classifier* bekerja. Setelah selesai, sistem akan menampilkan hasil klasifikasi beserta nilai akurasi yang didapat. Terakhir, dilakukan asosiasi teks untuk mengetahui informasi penting yang berguna bagi berbagai pihak. *Flowchart* algoritma sistem yang dibangun dapat dilihat pada Gambar 4.2.



Gambar 4.2 Flowchart sistem

4.2.2 Desain Fisik

Pada penelitian ini, tidak dilakukan perancangan tampilan antarmuka sistem yang akan dibuat karena sistem berupa analisis yang memuat kode *script* pada Bahasa pemrograman R. Dalam implementasinya, penulis menggunakan *software* RStudio dengan bantuan *package* sehingga sistem mampu berjalan dengan baik. Pada Gambar 4.3 berikut merupakan antarmuka dari *software* RStudio.



Gambar 4.3 Antarmuka *software* RStudio

BAB V

IMPLEMENTASI DAN HASIL

5.1 Implementasi

Implementasi merupakan penerapan terhadap cara kerja sistem yang telah dirancang berdasarkan analisis dan desain sistem yang sudah dibuat. Pada implementasi ini membahas tentang hasil implementasi sistem dan hasil pengujian sistem. Implementasi sistem adalah tahap di mana sistem telah diterapkan dan siap untuk digunakan. Bab ini akan membahas kode program maupun penjelasan dari proses *web scraping* dalam mengumpulkan data, proses *data preprocessing* dalam menormalisasi data, proses labelisasi data untuk pembobotan kata berdasarkan sentimen positif dan negatif, proses *machine learning* atau klasifikasi sentimen, proses evaluasi terhadap hasil klasifikasi, dan asosiasi teks.

5.1.1 Proses Web Scraping

Sebelum melakukan klasifikasi sentimen data ulasan pengguna aplikasi PLN Mobile, tahap pertama yang perlu dilakukan ialah mengumpulkan data melalui teknik *web scraping*. Pada tahap ini, bahasa pemrograman Python digunakan karena lebih efektif dalam mengumpulkan data secara cepat. Selain itu, *cloud environment* Google Colaboratory dan *library* Google Play Scraper digunakan dalam proses *web scraping* agar tidak membebani sumber daya komputer penulis. Adapun luaran yang diharapkan pada tahap ini berupa dataset ulasan dengan format *file* CSV karena mudah diolah dan dianalisis. Gambar 5.1 dan 5.2 berikut merupakan *source code* dari proses *web scraping* data ulasan aplikasi PLN Mobile.

```

1 !pip install google-play-scraper
2 import pandas as pd
3 from tqdm import tqdm
4 from google_play_scraper import Sort, reviews, app
5 app_id = ['com.icon.pln123']
6 app_reviews = []
7 for app in tqdm(app_id):
8     for score in list(range(1, 6)):
9         for sort_order in [Sort.MOST_RELEVANT, Sort.NEWEST]:
10             rvs, _ = reviews(
11                 app,
12                 lang = 'id',
13                 country = 'id',
14                 sort = sort_order,
15                 count= 400 if score == 3 else 400,
16                 filter_score_with = score
17             )
18             for r in rvs:
19                 r['sortOrder'] = 'most_relevant' if sort_order == Sort.MOST_RELEVANT else 'newest'
20                 r['appId'] = app
21             app_reviews.extend(rvs)

```

Gambar 5.1 Source code proses web scraping

Baris *script* ke-1 adalah instruksi untuk menginstal *library* Google Play Scraper di *cloud environment* Google Colaboratory. Baris *script* ke-2 sampai 4 adalah memanggil *library* yang dibutuhkan system antara lain Pandas, TQDM, dan Google Play Scraper. Baris *script* ke-5 dan 6 adalah mendefinisikan variabel ID aplikasi PLN Mobile dan variabel *array* untuk menampung data ulasan dari Google Play Store. Baris *script* ke-7 adalah instruksi perulangan untuk menampilkan progress *web scraping* dalam bentuk *progress bar*. Baris *script* ke-8 adalah instruksi perulangan agar mampu menampilkan data ulasan dengan *rating* bintang satu sampai lima. Baris *script* ke-9 sampai 17 adalah instruksi perulangan untuk proses *web scraping* yang mana didasarkan pada beberapa parameter seperti ID aplikasi, Bahasa, negara, penyortiran, jumlah ulasan, dan *rating*. Pada awalnya, data hasil *web scraping* dalam tipe data *list of character* yang mana terpisah, sehingga baris *script* ke-18 sampai 21 adalah instruksi perulangan untuk mengumpulkan setiap iterasi data hasil *web scraping* menjadi satu kesatuan *list of character*.

```

22 len(app_reviews)
23 df = pd.DataFrame(app_reviews)
24 df
25 df['score'].value_counts()
26 df.drop(df.loc[df['score']==3].index, inplace=True)
27 len(df)
28 df.drop(['reviewId', 'userImage', 'userName', 'score', 'thumbsUpCount',
29         'reviewCreatedVersion', 'at', 'replyContent', 'repliedAt',
30         'sortOrder', 'appId'], axis=1, inplace=True)
31 df.head()
32 df.to_csv("raw_dataset.csv", index=None, header=None)
33 from google.colab import files
34 files.download("raw_dataset.csv")

```

Gambar 5.2 Source code proses web scraping (lanjutan)

Baris *script* ke-22 adalah instruksi untuk menampilkan jumlah data hasil *web scraping*. Baris *script* ke-23 dan 24 adalah mengkonversi *list of character* menjadi *dataframe* dan menampilkan *dataframe* tersebut. Baris *script* ke-25 adalah menampilkan jumlah data pada masing-masing *value* dalam kolom *score*. Baris *script* ke-26 adalah instruksi untuk menghapus data dengan nilai *score* sama dengan tiga. Hal ini perlu dilakukan untuk menghindari ulasan atau komentar yang netral. Baris *script* ke-27 adalah instruksi untuk menampilkan jumlah data setelah dilakukan penghapusan beberapa data. Baris *script* ke-28 sampai 30 adalah menghapus beberapa kolom yang tidak diperlukan dalam pengolahan data. Baris *script* ke-31 adalah menampilkan lima data teratas dari *dataframe*. Baris *script* ke-32 adalah instruksi untuk mengkonversi *dataframe* menjadi *file* dengan format CSV. Baris *script* ke-33 dan 34 adalah instruksi untuk mendefinisikan *library* pengunduh *file* dan mengunduh *file* tersebut ke dalam computer lokal.

5.1.2 Proses Data Preprocessing

Sebelum melakukan labelisasi data pada ulasan pengguna aplikasi PLN Mobile, perlu dilakukan *data preprocessing*. Data ulasan yang diperoleh melalui teknik *web scraping* belum sepenuhnya siap digunakan untuk proses labelisasi data karena data masih belum terstruktur dengan baik dan terdapat banyak *noise*. Data masih memuat angka, tanda baca, *emoticon*, serta kata-kata yang kurang bermakna untuk dijadikan fitur. Maka dari itu, perlu dilakukan *data preprocessing* yang bertujuan untuk menyeragamkan bentuk kata, menghilangkan karakter-karakter selain huruf, dan mengurangi jumlah kosakata sehingga data menjadi mudah diolah. Gambar 5.3 sampai 5.5 berikut merupakan *source code* dari proses *data preprocessing* data ulasan aplikasi PLN Mobile.

```

1 library(tm)
2 library(tokenizers)
3 library(dplyr)
4 library(textclean)
5 library(stopwords)
6 library(parallel)
7 library(katadasaR)
8 setwd("E:/PROGRAM KP/")
9 dataset_mentah <- readLines("raw_dataset.csv")
10 dataset_mentah[1:10]
11 hasil_casefolding <- tolower(dataset_mentah)
12 hasil_casefolding[1:10]
13 hasil_cleaning1 <- gsub("[^\x20-\x7E]", "", hasil_casefolding)
14 hasil_cleaning1[1:10]
15 hasil_cleaning2 <- gsub("\\d", "", hasil_cleaning1)
16 hasil_cleaning2[1:10]
17 hasil_cleaning3 <- gsub("[[:punct:]]+", "", hasil_cleaning2)
18 hasil_cleaning3[1:10]
19 hasil_cleaning4 <- gsub("\\s+", " ", hasil_cleaning3)
20 hasil_cleaning4[1:10]

```

Gambar 5.3 Source code proses data preprocessing pertama

Baris *script* ke-1 sampai 7 adalah instruksi untuk memanggil *library* atau *package* yang dibutuhkan dalam membangun sistem meliputi TM, Tokenizers, DPLYR, TextClean, Stopwords, Parallel, dan katadasaR. Baris *script* ke-8 adalah mengatur *folder* kerja dalam komputer lokal. Baris *script* ke-9 dan 10 adalah membaca *dataset* dan menampilkan sepuluh baris data teratas. Baris *script* ke-11 dan 12 adalah instruksi untuk melakukan proses *case folding* dan menampilkan hasilnya. Baris *script* ke-13 dan 14 adalah instruksi untuk melakukan proses penghapusan karakter non-ASCII dan menampilkan hasilnya. Baris *script* ke-15 dan 16 adalah instruksi untuk melakukan proses penghapusan angka dan menampilkan hasilnya. Baris *script* ke-17 dan 18 adalah instruksi untuk melakukan proses penghapusan tanda baca dan menampilkan hasilnya. Baris *script* ke-19 dan 20 adalah instruksi untuk melakukan proses penghapusan spasi yang berlebihan dan menampilkan hasilnya.

```

23 kamus_slang <- read.csv("colloquial-indonesian-lexicon.csv", sep=",", header=TRUE)
24 hasil_normalisasi <- replace_internet_slang(hasil_cleaning4, slang=paste0("\\b",
25                                             kamus_slang$slang, "\\b"),
26                                             replacement=kamus_slang$formal, ignore.case=TRUE)
27 hasil_normalisasi[1:10]
28 stemming <- function(x){
29   paste(lapply(x, katadasaR), collapse = " ")
30 }
31 hasil_stemming <- lapply(tokenize_words(hasil_normalisasi[]), stemming)
32 hasil_stemming[1:10]
33 hasil_tokenizing <- tokenize_words(hasil_stemming)
34 head(hasil_tokenizing, 20)
35 hasil_tokenizing_char <- as.character(hasil_tokenizing)
36 kamus_stopword <- readLines("stopwords.txt")
37 hasil_stopword <- tokenize_words(hasil_tokenizing_char, stopwords=kamus_stopword)
38 hasil_stopword[11:20]
39 hasil_stopword_char <- sapply(hasil_stopword, toString)
40 hasil_stopword_char[1:10]
41 hasil_stopword_bersih1 <- gsub("[:punct:]]+", "", hasil_stopword_char)
42 hasil_stopword_bersih1[1:10]
43 hasil_stopword_bersih2 <- gsub(" *\\b[:alpha:]]{1,3}\\b *", "", hasil_stopword_bersih1)
44 hasil_stopword_bersih2[1:10]

```

Gambar 5.4 Source code proses data preprocessing kedua

Baris *script* ke-23 sampai 27 adalah instruksi untuk melakukan proses *spell normalization* dan menampilkan hasilnya. Baris *script* ke-28 sampai 32 adalah instruksi untuk melakukan proses *stemming* dan menampilkan hasilnya. Baris *script* ke-33 sampai 35 adalah instruksi untuk melakukan proses *tokenizing*, menampilkan hasilnya, dan mengkonversi dari tipe data *matrix* ke tipe data *character*. Baris *script* ke-36 sampai 38 adalah instruksi untuk membaca *stoplist*, melakukan proses *filtering*, dan menampilkan hasilnya. Baris *script* ke-39 dan 40 adalah instruksi untuk melakukan proses konversi dari tipe data *matrix* ke tipe data *string* dan menampilkan hasilnya. Baris *script* ke-41 sampai 44 adalah instruksi untuk melakukan proses penghapusan kembali terhadap tanda baca dan kata dengan satu sampai tiga huruf saja, kemudian menampilkan hasilnya.

```

45 dataset_corpus <- Corpus(VectorSource(hasil_stopword_bersih2))
46 df <- data.frame(text=unlist(sapply(dataset_corpus, `[`)), stringsAsFactors=FALSE)
47 View(df)
48 df[df==""] <- NA
49 sum(is.na(df))
50 dataset_hapusna <- na.omit(df) # hapus NA row
51 dataset_urut <- dataset_hapusna %>%
52   arrange(text)
53 dataset_bersih <- distinct(dataset_urut, text, .keep_all=TRUE)
54 View(dataset_bersih)
55 write.csv(dataset_bersih, "dataset_bersih.csv")
56 save.image(file = "program_KP.RData")

```

Gambar 5.5 Source code proses data preprocessing ketiga

Baris *script* ke-45 adalah instruksi untuk mengkonversi tipe data *string* menjadi *corpus*. Baris *script* ke-46 dan 47 adalah untuk mengkonversi *corpus* menjadi *dataframe* dan menampilkan hasilnya. Baris *script* ke-48 adalah untuk mengisi baris data yang kosong dengan tulisan NA (*Not Available*). Baris *script* ke-49 adalah untuk menghitung jumlah baris data yang kosong atau yang telah diberi tulisan NA. Baris *script* ke-50 adalah untuk melakukan penghapusan terhadap baris data yang kosong. Baris *script* ke-51 dan 52 adalah untuk mengurutkan ulasan berdasarkan huruf awal yaitu dimulai dari huruf a sampai huruf z. Baris *script* ke-53 adalah untuk melakukan penghapusan terhadap baris data yang memiliki kata duplikat atau memiliki kemiripan dengan baris data lainnya. Baris *script* ke-54 adalah untuk menampilkan data ulasan hasil proses *data preprocessing*. Baris *script* ke-55 dan 56 adalah untuk menyimpan data ulasan ke komputer lokal dalam bentuk *file* CSV dan menyimpan *session* dari program yang telah dijalankan di RStudio.

5.1.3 Proses Labelisasi Data

Klasifikasi merupakan teknik *machine learning* yang termasuk ke dalam pembelajaran terawasi (*supervised learning*), sehingga dibutuhkan pelabelan kelas sentimen terhadap data ulasan. Pada penelitian ini, pelabelan klasifikasi yang dilakukan pada analisis sentimen dibagi menjadi dua kelas sentimen yakni sentimen positif dan sentimen negatif. Berdasarkan rating yang diberikan pengguna dapat diketahui gambaran umum dari penilaian pengguna terhadap aplikasi Detikcom. Maka dari itu, ulasan dengan rating 1 dan rating 2 dapat dilabelkan ke dalam kelas negatif, sedangkan ulasan dengan rating 4 dan rating 5 dapat dilabelkan ke dalam positif. Gambar 5.6 dan 5.7 berikut merupakan *source code* dari proses labelisasi data ulasan aplikasi PLN Mobile.

```

1 library(stringr)
2 library(ggplot2)
3 library(ROSE)
4 setwd("E:/PROGRAM KP/")
5 dataset_siap <- read.csv("dataset_bersih.csv", header=TRUE)
6 positif <- scan("list-of-positive-words.txt", what="character", comment.char=";")
7 negatif <- scan("list-of-negative-words.txt", what="character", comment.char=";")
8 score.sentiment = function(myData, kata.positif, kata.negatif, .progress='none')
9 {
10   require(plyr)
11   scores = laply(myData, function(kalimat, kata.positif, kata.negatif) {
12     list.kata = str_split(kalimat, '\\s+')
13     kata2 = unlist(list.kata)
14     positif.matches = match(kata2, kata.positif)
15     negatif.matches = match(kata2, kata.negatif)
16     positif.matches = !is.na(positif.matches)
17     negatif.matches = !is.na(negatif.matches)
18     score = sum(positif.matches) - sum(negatif.matches)
19     return (score)
20   }, kata.positif, kata.negatif, .progress=.progress)
21   scores.df = data.frame(score=scores, text=myData)
22   return (scores.df)
23 }
24 hasil_skoring = score.sentiment(dataset_siap$text, positif, negatif)
25 View(hasil_skoring)

```

Gambar 5.6 Source code proses labelisasi data

Baris *script* ke-1 sampai 3 adalah instruksi untuk memanggil *library* atau *package* yang dibutuhkan dalam membangun sistem meliputi stringR, ggplot2, dan ROSE. Baris *script* ke-4 adalah mengatur *folder* kerja dalam komputer lokal. Baris *script* ke-5 adalah membaca *dataset* hasil proses *data preprocessing*. Baris *script* ke-6 dan 7 adalah membaca kamus kosakata berbahasa Indonesia yang telah dikelompokkan ke dalam kelas positif dan negatif. Baris *script* ke-8 sampai 23 adalah instruksi untuk mendefinisikan *function* pembobotan teks. Baris *script* ke-24 adalah instruksi untuk memberikan bobot nilai pada teks ulasan. Baris *script* ke-25 adalah instruksi untuk menampilkan hasil pembobotan teks.

```

26 hasil_skoring$sentiment <- ifelse(hasil_skoring$score < 0, "Negatif","Positif")
27 View(hasil_skoring)
28 table(hasil_skoring$sentiment)
29 ggplot(hasil_skoring, aes(x=factor(sentiment), fill=factor(sentiment))) +
30   geom_bar(aes(y = (..count..)/sum(..count..))) +
31   scale_y_continuous(labels=percent) + xlab("Sentiment") +
32   ylab("Persentase") + ggtitle("Tabel Distribusi Kolom Sentiment") + theme_grey(base_size=10)
33 hasil_oversampling <- ovun.sample(sentiment~., data=hasil_skoring,
34                                   method="over", p=0.5, seed=1)$data
35 ggplot(hasil_oversampling, aes(x=factor(sentiment), fill=factor(sentiment))) +
36   geom_bar(aes(y = (..count..)/sum(..count..))) +
37   scale_y_continuous(labels=percent) + xlab("Sentiment") +
38   ylab("Persentase") + ggtitle("Tabel Distribusi Kolom Sentiment") + theme_grey(base_size=10)
39 dataset_terlabel <- hasil_oversampling[c(2,1,3)]
40 View(dataset_terlabel)
41 data_positif <- dataset_terlabel %>%
42   filter(sentiment == "Positif")
43 View(data_positif)
44 data_negatif <- dataset_terlabel %>%
45   filter(sentiment == "Negatif")
46 View(data_negatif)
47 write.csv(dataset_terlabel, file="dataset_terlabel.csv")
48 write.csv(data_positif, file="dataset_positif.csv")
49 write.csv(data_negatif, file="dataset_negatif.csv")
50 save.image(file = "program_KP.RData")

```

Gambar 5.7 Source code proses labelisasi data (lanjutan)

Baris *script* ke-26 dan 27 adalah instruksi untuk melakukan proses labelisasi data dan menampilkan hasilnya. Baris *script* ke-28 adalah instruksi untuk menghitung jumlah baris data pada setiap kelas sentimen. Baris *script* ke-29 sampai 32 adalah instruksi untuk membuat visualisasi *plot* dari jumlah baris data pada setiap kelas sentimen. Baris *script* ke-33 dan 34 adalah instruksi untuk melakukan proses *oversampling* guna menyeimbangkan jumlah baris data pada setiap kelas sentimen. Baris *script* ke-35 dan 38 adalah instruksi untuk membuat visualisasi *plot* dari data hasil proses *oversampling*. Baris *script* ke-39 dan 40 adalah untuk mengubah posisi kolom pada tabel hasil proses labelisasi data. Baris *script* ke-41 sampai 43 adalah untuk menyortir *dataset* hanya dengan kosakata positif. Baris *script* ke-44 sampai 46 adalah untuk menyortir *dataset* hanya dengan kosakata negatif. Baris *script* ke-47 sampai 49 adalah untuk menyimpan *dataset* hasil proses labelisasi data, *dataset* dengan kosakata positif, dan *dataset* dengan kosakata negatif. Baris *script* ke-50 adalah untuk menyimpan *session* dari program yang telah dijalankan di RStudio.

5.1.4 Proses Klasifikasi Sentimen

Proses klasifikasi dilakukan dengan menggunakan data latih dan data uji dalam proses pembuatan *machine learning*. Model *machine learning* dibuat dengan

melakukan pelatihan pada data latih yang selanjutnya dapat diujikan menggunakan data uji. Pengujian model dilakukan untuk mengetahui tingkat keakurasian model dan sejauh mana model dapat melakukan klasifikasi dengan benar. Dalam membangun sistem dengan *machine learning*, terdapat beberapa metode algoritma yang dapat digunakan. Metode algoritma yang digunakan dalam penelitian ini adalah *Naïve Bayes Classifier*. Adapun *source code* dari proses klasifikasi sentimen data ulasan aplikasi PLN Mobile terdapat pada Gambar 5.8 sampai 5.10.

```

1 library(RTextTools)
2 library(e1071)
3 library(caret)
4 setwd("C:/Users/ALFIAN HIDAYATULLOH/Videos/")
5 dataset_final <- read.csv("dataset_terlabel.csv", header=TRUE)[ , c("text","score","sentiment")]
6 set.seed(100)
7 df_rows <- sample(nrow(dataset_final))
8 my_df <- dataset_final[df_rows, ]
9 my_df <- dataset_final[df_rows, ]
10 View(my_df)
11 corpus_final <- Corpus(VectorSource(my_df$text))
12 inspect(corpus_final[1:10])
13 doctm <- DocumentTermMatrix(corpus_final)
14 inspect(doctm[1:10, 11:20])
15 #Partitioning (70:30)
16 #my_df.train <- my_df[0:2082,]
17 #my_df.test <- my_df[2083:2973,]
18 #doctm.train <- doctm[0:2082,]
19 #doctm.test <- doctm[2083:2973,]
20 #corpus_final.train <- corpus_final[0:2082]
21 #corpus_final.test <- corpus_final[2083:2973]
22 #Partitioning (75:25)
23 #my_df.train <- my_df[0:2231,]
24 #my_df.test <- my_df[2232:2973,]
25 #doctm.train <- doctm[0:2231,]
26 #doctm.test <- doctm[2232:2973,]
27 #corpus_final.train <- corpus_final[0:2231]
28 #corpus_final.test <- corpus_final[2232:2973]

```

Gambar 5.8 *Source code* proses klasifikasi sentimen pertama

Baris *script* ke-1 sampai 3 adalah instruksi untuk memanggil *library* atau *package* yang dibutuhkan dalam membangun sistem meliputi RtextTools, e1071 (*Naïve Bayes*), dan caret. Baris *script* ke-4 adalah mengatur *folder* kerja dalam komputer lokal. Baris *script* ke-5 adalah membaca *dataset* hasil proses labelisasi data. Baris *script* ke-6 adalah mendefinisikan angka random agar hasil dari proses distribusi persebaran data tetap sama meskipun dilakukan berulang-ulang. Baris *script* ke-7 sampai 10 adalah instruksi untuk melakukan proses distribusi persebaran data dan menampilkan hasilnya. Baris *script* ke-11 dan 12 adalah instruksi untuk mengkonversi *dataset* ulasan menjadi *corpus* dan menampilkan hasilnya. Baris *script* ke-13 dan 14 adalah instruksi untuk mengkonversi *corpus* menjadi *matrix* kosakata tiap ulasan dan menampilkan hasilnya. Baris *script* ke-15 sampai 21 adalah untuk membagi *dataset* ulasan, *matrix* kosakata, dan *corpus* data

menjadi data latih dan data uji dengan proporsi 70% dan 30%. Baris *script* ke-22 sampai 28 adalah untuk membagi *dataset* ulasan, *matrix* kosakata, dan *corpus* data menjadi data latih dan data uji dengan proporsi 75% dan 25%.

```

29 #Partitioning (80:20)
30 #my_df.train <- my_df[0:2379,]
31 #my_df.test <- my_df[2380:2973,]
32 #doctm.train <- doctm[0:2379,]
33 #doctm.test <- doctm[2380:2973,]
34 #corpus_final.train <- corpus_final[0:2379]
35 #corpus_final.test <- corpus_final[2380:2973]
36 #Partitioning (85:15)
37 #my_df.train <- my_df[0:2528,]
38 #my_df.test <- my_df[2529:2973,]
39 #doctm.train <- doctm[0:2528,]
40 #doctm.test <- doctm[2529:2973,]
41 #corpus_final.train <- corpus_final[0:2528]
42 #corpus_final.test <- corpus_final[2529:2973]
43 #Partitioning (90:10)
44 my_df.train <- my_df[0:2677,]
45 my_df.test <- my_df[2678:2973,]
46 doctm.train <- doctm[0:2677,]
47 doctm.test <- doctm[2678:2973,]
48 corpus_final.train <- corpus_final[0:2677]
49 corpus_final.test <- corpus_final[2678:2973]
50 write.csv(my_df.train, file = "training_data.csv")
51 write.csv(my_df.test, file = "testing_data.csv")
52 dim(doctm.train)
53 fivefreq <- findFreqTerms(doctm.train, 1)
54 length(fivefreq)

```

Gambar 5.9 Source code proses klasifikasi sentimen kedua

Baris *script* ke-29 sampai 35 adalah untuk membagi *dataset* ulasan, *matrix* kosakata, dan *corpus* data menjadi data latih dan data uji dengan proporsi 80% dan 20%. Baris *script* ke-36 sampai 42 adalah untuk membagi *dataset* ulasan, *matrix* kosakata, dan *corpus* data menjadi data latih dan data uji dengan proporsi 85% dan 15%. Baris *script* ke-43 sampai 49 adalah untuk membagi *dataset* ulasan, *matrix* kosakata, dan *corpus* data menjadi data latih dan data uji dengan proporsi 90% dan 10%. Baris *script* ke-50 dan 51 adalah instruksi untuk menyimpan *dataset* ulasan yang berupa data latih dan data uji ke komputer lokal. Baris *script* ke-52 adalah instruksi untuk menghitung dimensi (baris dan kolom) dari data latih. Baris *script* ke-53 adalah untuk menemukan kata-kata dengan frekuensi tinggi dari data latih. Baris *script* ke-54 adalah menghitung jumlah kata dengan frekuensi tinggi.

```

55 doctm.train.nb <- DocumentTermMatrix(corpus_final.train, control=list(dictionary = fivefreq))
56 dim(doctm.train.nb)
57 doctm.test.nb <- DocumentTermMatrix(corpus_final.test, control=list(dictionary = fivefreq))
58 dim(doctm.test.nb)
59 convert_countNB <- function(x){
60   y <- ifelse(x > 0, 1, 0)
61   y <- factor(y, levels=c(0, 1), labels=c("No", "Yes"))
62   y
63 }
64 TrainNB = read.csv("training_data.csv", header=TRUE)[ , c("text","score","sentiment")]
65 TestNB = read.csv("testing_data.csv", header=TRUE)[ , c("text","score","sentiment")]
66 trainNB <- apply(doctm.train.nb, 2, convert_countNB)
67 testNB <- apply(doctm.test.nb, 2, convert_countNB)
68 classifier <- naiveBayes(trainNB, my_df.train$sentiment, laplace=1)
69 pred <- predict(classifier, testNB)
70 mytable <- table("Predictions"=pred, "Actual"=my_df.test$sentiment)
71 conf.mat <- confusionMatrix(mytable)
72 conf.mat
73 conf.mat$byClass
74 conf.mat$overall
75 conf.mat$overall['Accuracy']*100
76 save.image(file = "program KP.RData")

```

Gambar 5.10 Source code proses klasifikasi sentimen ketiga

Baris *script* ke-55 dan 56 adalah instruksi untuk membentuk data latih berupa dokumen matriks dari kata-kata yang memiliki frekuensi kemunculan tinggi, serta menghitung dimensi datanya. Baris *script* ke-57 dan 58 adalah instruksi untuk membentuk data uji berupa dokumen matriks dari kata-kata yang memiliki frekuensi kemunculan tinggi, serta menghitung dimensi datanya. Baris *script* ke-59 sampai 63 adalah untuk mendefinisikan *function* konversi perhitungan skor sentimen kata yang semula angka biner menjadi tipe data *boolean*. Baris *script* ke-64 dan 65 adalah untuk membaca data latih dan data uji yang disimpan sebelumnya. Baris *script* ke-66 dan 67 adalah untuk menerapkan *function* konversi terhadap dokumen matriks kosakata yang telah didefinisikan sebelumnya, angka 2 berarti *function* diterapkan pada setiap kolom. Baris *script* ke-68 adalah untuk mendefinisikan sistem pengklasifikasi menggunakan *Naïve Bayes Classifier* dengan fitur *laplace* (pembangkit probabilitas nol) aktif. Baris *script* ke-69 adalah untuk memprediksi klasifikasi sentimen kata ke dalam kelas positif atau negatif. Baris *script* ke-70 sampai 75 adalah untuk menampilkan akurasi hasil klasifikasi menggunakan *Naïve Bayes Classifier*. Baris *script* ke-76 adalah untuk menyimpan *session* dari program yang telah dijalankan di RStudio.

5.1.5 Proses Asosiasi Teks

Asosiasi teks diperoleh dengan menggunakan pendekatan nilai korelasi terhadap masing-masing kata terhadap kemungkinan suatu kata dibahas bersamaan

dengan kata lainnya. Berdasarkan kata-kata yang paling banyak muncul, dapat diperoleh asosiasi antarkata pada masing-masing kelas sentimen secara bersamaan guna memperoleh informasi. Adapun *source code* dari proses asosiasi teks untuk data ulasan positif aplikasi PLN Mobile terdapat pada Gambar 5.11 sampai 5.13.

```

1 library(RColorBrewer)
2 library(wordcloud)
3 setwd("E:/PROGRAM KP/")
4 dataset_positif <- read.csv("dataset_positif.csv", header=TRUE)[ , c("text","score","sentiment")]
5 corpus_positif <- Corpus(VectorSource(dataset_positif$text))
6 corpus_positif2 <- tm_map(corpus_positif, removeWords, c("kasih","terima","tugas","bintang","keluh","alamibatulucin",
7 "tolong","terimakasih","sekali","banget","sahari","close",
8 "nomor","kali","kerja","selesai","laku","padam",
9 "kalo","ganggu","maaf","aplikasimobile","mati",
10 "tingkat","info","lapang","layananan","tulisi",
11 "download","fast","berik","lumlayananan","moga",
12 "mobile","rating","nama","malam","trimakasih",
13 "live","april","januari","call","ribet","habis",
14 "adayg","besokmati","dahdi","diaplikasimemangnya",
15 "dimanaterpaksa","jugakalo","termakasih","cobafliip",
16 "closed","endingnya","batalnomor","kalocontoh"))
17 corpus_positif3 <- tm_map(corpus_positif2, gsub, pattern="ayan", replacement="layananan")
18 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="langgan", replacement="langganan")
19 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="hubung", replacement="kontak")
20 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="layananan", replacement="jasa")
21 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="muas", replacement="puas")
22 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="responnya", replacement="respon")
23 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="bagusjasa", replacement="jasa")
24 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="dayaupgrade", replacement="upgrade")
25 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="internetuntuk", replacement="internet")
26 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="tersebuttolong", replacement="tolong")
27 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="hasilstatus", replacement="status")
28 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="hasiltombol", replacement="tombol")
29 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="listrikpln", replacement="listrik")
30 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="fiturmobile", replacement="fitur")

```

Gambar 5.11 *Source code* proses asosiasi teks pertama

Baris *script* ke-1 dan 2 adalah instruksi untuk memanggil *library* atau *package* yang dibutuhkan dalam membangun sistem meliputi RcolorBrewer dan wordcloud. Baris *script* ke-3 adalah mengatur *folder* kerja dalam komputer lokal. Baris *script* ke-4 adalah membaca *dataset* hasil proses labelisasi data. Baris *script* ke-5 adalah mengkonversi *dataset* menjadi *corpus*. Baris *script* ke-16 sampai 17 adalah instruksi untuk melakukan proses penghapusan beberapa kata (*noise*) yang tak diperlukan. Baris *script* ke-17 sampai 30 adalah instruksi untuk melakukan pergantian bentuk kata dari yang awalnya *slang word* menjadi ternormalisasi.

```

31 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="belistrukjadi",replacement="struk")
32 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="bagusjasa",replacement="jasa")
33 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="listrikhabis",replacement="listrik")
34 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="meteransudah",replacement="meter")
35 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="tokenhari",replacement="token")
36 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="auto",replacement="otomatis")
37 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="keperayaanmobile",replacement="percaya")
38 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="rumahstabil",replacement="stabil")
39 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="aplikasiasli",replacement="asli")
40 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="tokped",replacement="tokopedia")
41 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="ganggumobiletercloseselesai",replacement="ganggu close selesai")
42 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="meresponselesai",replacement="respon")
43 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="notifikasiterclose",replacement="notifikasi")
44 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="teleponsaran",replacement="saran")
45 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="ecek",replacement="periksa")
46 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="ekstrapenuh",replacement="ekstra")
47 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="padamproses",replacement="proses")
48 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="statussuruh",replacement="status")
49 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="kalobebasin",replacement="bebas")
50 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="tokentransaksi",replacement="transaksi")
51 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="kalomobile",replacement="mobile")
52 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="ranting",replacement="rating")
53 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="alhamdulillahaku",replacement="alhamdulillah")
54 corpus_positif3 <- tm_map(corpus_positif3, gsub, pattern="servicenyanam",replacement="service")
55 tdm <- TermDocumentMatrix(corpus_positif3)
56 matrix <- as.matrix(tdm)
57 vector <- sort(rowSums(matrix), decreasing=TRUE)
58 data_frame <- data.frame(word = names(vector), freq=vector)
59 head(data_frame, 50)
60 set.seed(123)
61 wordcloud(words = data_frame$word, freq = data_frame$freq, min.freq = 1,
62           max.words=50, random.order=FALSE, rot.per=0.35,
63           colors=brewer.pal(8, "Set2"))

```

Gambar 5.12 Source code proses asosiasi teks kedua

Baris *script* ke-31 sampai 54 adalah instruksi untuk melakukan pergantian bentuk kata dari yang awalnya *slang word* menjadi ternormalisasi. Baris *script* ke-55 adalah untuk mengkonversi *corpus* menjadi *Term Document Matrix* yang berisi kumpulan kata pada sentimen positif. Baris *script* ke-56 dan 57 adalah untuk mengkonversi *Term Document Matrix* menjadi bentuk matriks dan memvektorisasi matriks tersebut. Baris *script* ke-58 dan 59 adalah instruksi untuk mengkonversi vektor menjadi *dataframe* dan menampilkan hasilnya. Baris *script* ke-60 adalah mendefinisikan angka random agar hasil dari proses distribusi persebaran data tetap sama meskipun dilakukan berulang-ulang. Baris *script* ke-61 sampai 63 adalah intruksi untuk membuat visualisasi data berupa *wordcloud* yang berisi kumpulan kata positif dengan frekuensi kemunculan kata tertinggi.

```

64 findFreqTerms(tdm, lowfreq = 10)
65 myvector <- as.list(findAssocs(tdm, terms=c("aplikasi","transaksi","layanan",
66                                           "cepat","harga"),
67                                           corlimit = c(0.15,0.15,0.15,0.15,0.15)))
68 myvector
69 myv <- as.list(findAssocs(tdm, terms =c("aplikasi"),
70                               corlimit = c(0.15)))
71 View(myv$aplikasi)
72 k<-barplot(data_frame[1:10,]$freq, las = 2, names.arg = data_frame[1:10,]$word,
73           cex.axis=1.2,cex.names=1.2,
74           main = "Kata Positif yang Paling Sering Muncul",
75           ylab = "Frekuensi",col = terrain.colors(20))
76 termFrequency <- rowSums(as.matrix(tdm))
77 termFrequency <- subset(termFrequency, termFrequency>101)
78 text(k, sort(termFrequency, decreasing = TRUE),
79      labels=sort(termFrequency, decreasing = T),pch=6, cex=1)
80 save.image(file = "program_KP.RData")

```

Gambar 5.13 Source code proses asosiasi teks ketiga

Baris *script* ke-64 adalah instruksi untuk menampilkan kata-kata dalam dokumen matriks yang memiliki frekuensi kemunculan tinggi, yakni dengan frekuensi minimal 10 kali kemunculan. Baris *script* ke-65 sampai 68 adalah instruksi untuk mencari korelasi antarkata (yang telah didefinisikan) dengan nilai korelasi minimal 0,15 dan kemudian menampilkan hasilnya. Baris *script* ke-69 sampai 71 adalah untuk mencari korelasi pada satu kata (yang telah didefinisikan) dan menampilkan hasilnya. Baris *script* ke-72 sampai 79 adalah untuk memvisualisasikan data dalam bentuk *barplot* untuk kata-kata positif dengan frekuensi tertinggi. Baris *script* ke-80 adalah untuk menyimpan *session* dari program yang telah dijalankan di RStudio.

5.2 Hasil

5.2.1 Hasil Proses Web Scraping

Pada penelitian ini, dilakukan pengambilan data berupa ulasan pengguna aplikasi PLN Mobile pada situs Google Play Store. Pengambilan data dilakukan dengan teknik *web scraping* menggunakan *library* dalam bahasa pemrograman Python. Sebenarnya terdapat beragam *library* yang dapat digunakan untuk mengambil data pada situs *online*. Namun penulis menggunakan *library* Google Play Scraper karena dapat digunakan untuk mengekstraksi data pada situs *online* secara gratis. Adapun pada penelitian ini, digunakan *library* Google Play Scraper versi 1.0.0. Data yang diekstraksi kemudian dapat diekspor menjadi *spreadsheet* Microsoft Excel (CSV) dan diunduh ke dalam komputer lokal. Contoh hasil proses *web scraping* menggunakan *library* Google Play Scraper seperti pada Gambar 5.14.

	content
0	Program nambah id pelanggan masih belum bisa saya input, ket: selalu saja, sistem sedang sibuk, silakan coba beberapa saat lagi V001. Kenapa keterangan tiket nomer tiket 1183886 sudah terselesaikan/ditutup, sedangkan masalah masih belum terselesaikan (semua instruksi dari balasan email sdh saya dicoba). Ini balasan email dari help desk Kepada Yth. Bapak/Ibu, Keluhan Bapak/Ibu dengan Nomor Tiket 1183886 terkait NEW PLN MOBILE KELUHAN telah di selesaikan
1	Saya melakukan pengaduan pukul 08.28 wib terkait matinya lampu dirumah saya sedangkan dirumah tetangga tidak. Riwayat status di aplikasi penugasan regu pukul 08.31 wib dan dalam perjalanan pukul 10.52 wib. Sampai sekarang pukul 12.18 petugas pln belum juga datang.
2	Ajukan penambahan daya pakai promo 202 ribu yang lewat aplikasi ini. Udah 2 minggu gak ada penanganan. Bikin pengaduan statusnya langsung selesai tanpa tindak lanjut. Ini lah potret kalau BUMN masih diisi oleh orang-orang Old Mind dan Korup. Gak bakalan maju Negara kita ini. Warung kecil jualan token listrik bisa untung. PLN jualan listrik tp Utangnya nambah. Hancur hancur.
3	Kenapa untuk yg mau buat pasang baru lokasi hanya di Jawa timur? Tolong dong penjelasannya, saya download apps ini karena mau pasang baru, dan saya sudah coba dr website, tp gk bs makanya saya download apps ini eh malah makin gk bs tolong dong diperbaiki apps ini, terima kasih
4	Parah, beli token melalui aplikasi sudah 5 hari keterangannya masih diproses. Sudah melakukan PENGADUAN melalui aplikasi juga, tidak ada tindak lanjut tapi laporan saya keterangannya sudah selesai. Sudah tiga kali pelaporan dan tidak ada kemajuan sama sekali.

Gambar 5.14 Hasil proses *web scraping*

5.2.2 Hasil Proses Data Preprocessing

Pada tahap *data preprocessing*, akan dilakukan pembersihan data menggunakan metode *text mining*. Beberapa tahap yang akan dilakukan di antaranya meliputi *case folding*, *cleaning*, *spelling normalization*, *stemming*, *tokenizing*, dan *filtering* yang akan dijelaskan sebagai berikut.

a) *Case Folding*

Case folding adalah proses penyeragaman bentuk huruf di mana dalam proses ini hanya menerima huruf latin antara “a” sampai “z”. Karakter lain selain huruf dianggap sebagai *delimiter* sehingga karakter tersebut akan dihapus dari *dataset*. Kemudian penyeragaman dilakukan dengan mengubah isi dokumen menjadi huruf kecil secara keseluruhan (dari “a” sampai dengan “z”). Hal ini bertujuan agar kata yang ditulis dengan huruf awal kapital dan huruf nonkapital tidak terdeteksi memiliki arti yang berbeda. Implementasi *case folding* dapat dilihat salah satu contoh data ulasan aplikasi PLN Mobile pada Tabel 5.1 berikut.

Tabel 5.1 Hasil proses *case folding*

Data input	Data output
P elayanan live chat tidak maksimal cs cepat kali meninggalkan layanan belum sampai 3 menit sdh tidak bisa lagi. D i tambah karyawan yg profesional klu karyawan banyak berarti waktunya bisa di tambah minimal 5 menit untuk menulis/mengetik permasalahannya.	p elayanan live chat tidak maksimal cs cepat kali meninggalkan layanan belum sampai 3 menit sdh tidak bisa lagi. d i tambah karyawan yg profesional klu karyawan banyak berarti waktunya bisa di tambah minimal 5 menit untuk menulis/mengetik permasalahannya.

b) *Cleaning*

Pada tahap ini, dilakukan pembersihan ulasan dari simbol ataupun karakter yang tidak diperlukan seperti angka, singkatan, kata yang tidak diperlukan, tanda baca, dan lain sebagainya. Tabel 5.2 sampai 5.5 berikut merupakan salah satu contoh ulasan hasil dari proses *cleaning*.

- Menghapus karakter non-ASCII

Tabel 5.2 Hasil proses *cleaning* pertama

Data input	Data output
kenapa pas ambil foto ktp tidak bisa?? sudah di tekan\"\", ktp sudah di paskan dgn bingkai ttp tdk bisa. mohon di perbaiki secepatnya, karna sangat butuh pelayanan ini terimakasih öŸ™\u008f\	kenapa pas ambil foto ktp tidak bisa?? sudah di tekan\"\", ktp sudah di paskan dgn bingkai ttp tdk bisa. mohon di perbaiki secepatnya, karna sangat butuh pelayanan ini terimakasih \

- Menghapus angka

Tabel 5.3 Hasil proses *cleaning* kedua

Data input	Data output
pelayanan live chat tidak maksimal cs cepat kali meninggalkan layanan belum sampai 3 menit sdh tidak bisa lagi. di tambah karyawan yg profesional klu karyawan banyak berarti waktunya bisa di tambah minimal 5 menit untuk menulis/mengetik permasalahannya.	pelayanan live chat tidak maksimal cs cepat kali meninggalkan layanan belum sampai menit sdh tidak bisa lagi. di tambah karyawan yg profesional klu karyawan banyak berarti waktunya bisa di tambah minimal menit untuk menulis/mengetik permasalahannya.

- Menghapus tanda baca

Tabel 5.4 Hasil proses *cleaning* ketiga

Data input	Data output
pengajuan keluhan listrik mati... d aplikasi tertulis perjalanan, tp saat di lacak lokasi tidak berubah, d aplikasi tertulis nyala sementara, tp kenyataanya masih padam, aplikasi apaan ini !!!!!!!\	pengajuan keluhan listrik mati d aplikasi tertulis perjalanan tp saat di lacak lokasi tidak berubahd aplikasi tertulis nyala sementara tp kenyataanya masih padam aplikasi apaan ini

- Menghapus spasi yang berlebihan

Tabel 5. 5 Hasil proses *cleaning* keempat

Data input	Data output
gak bisa masuk terus ■ katanya nomor ponsel sudah terdaftar gunakan nomor ponsel lain ■ kan aneh.	gak bisa masuk terus katanya nomor ponsel sudah terdaftar gunakan nomor ponsel lain kan aneh.

c) *Spelling Normalization*

Pada bagian *spelling normalization* berguna untuk melakukan perbaikan kata-kata yang disingkat maupun salah ejaan dengan bentuk tertentu yang memiliki maksud yang sama. Sebagai contoh pada kata “tidak”, memiliki banyak bentuk penulisan yaitu “tak”, “enggak”, “tdak”, “tdk”, dan sebagainya. Dalam penelitian ini, proses *spelling normalization* dilakukan dengan bantuan kamus *slang word* yang telah dikurasi oleh Nikmatun Aliyah Salsabila dan timnya. Adapun salah satu contoh perbaikan kata gaul atau *slang word* pada ulasan seperti terlihat pada Tabel 5.6 berikut.

Tabel 5.6 Hasil proses *spelling normalization*

Data input	Data output
ini kenapa ya saya beli token tp gak dapat nomor tokennya sampai bolak balik restart hp trs skrg mau login malah gak bisa aplikasinya msh banyak bug sepertinya tlg bantu tokennya sekalian login itu kasih pilihan pakai nomor hp hg .	ini kenapa ya saya beli token tapi tidak dapat nomor tokennya sampai bolak balik restart hp terus sekarang mau login malah tidak bisa aplikasinya masih banyak bug sepertinyat tolong bantu tokennya sekalian login itu kasih pilihan pakai nomor hp juga .

d) *Stemming*


Stemming merupakan tahap yang dilakukan untuk mengubah kata berimbuhan pada ulasan menjadi kata dasar. Pada tahap ini, dilakukan penghapusan imbuhan pada kata hasil dari proses *spelling normalization*, baik yang mengandung imbuhan induktif maupun deduktif. *Stemming* dapat dilakukan pada bahasa pemrograman R menggunakan *package* bernama *katadasaR*. Adapun Tabel 5.7 berikut merupakan salah satu contoh ulasan dari hasil proses *stemming*.

Tabel 5.7 Hasil proses *stemming*

Data input	Data output
pengaduan di aplikasi selesai tapi di lapangan enggak ada jadi bingung mau isi token ada kata periksa terus kebocoran juga sudah diperiksa dan dibetulkan tetap saja tiap mau isi token selalu ada kata periksa pakai pengaduan di pln mobile lambat lebih cepat dulu pakai wa langsung ada solusi bingung sendiri deh.	adu di aplikasi selesai tapi di lapang enggak ada jadi bingung mau isi token ada kata periksa terus bocor juga sudah periksa dan betul tetap saja tiap mau isi token selalu ada kata periksa pakai adu di pln mobile lambat lebih cepat dulu pakai wa langsung ada solusi bingung sendiri deh.

e) *Tokenizing*

Tokenizing adalah proses memisahkan kata pada sebuah kalimat menjadi kata-kata yang independen. *Tokenizing* dilakukan untuk mendapatkan *token* atau potongan kata yang akan menjadi entitas yang memiliki nilai dalam penyusunan matriks pada proses selanjutnya. *Tokenizing* dapat memudahkan proses perhitungan keberadaan kata tersebut dalam kalimat ataupun untuk menghitung frekuensi kemunculan kata tersebut dalam *corpus*. Contoh proses *tokenizing* ditunjukkan pada Gambar 5.15 berikut.

adu di aplikasi selesai tapi di lapang enggak ada jadi bingung mau isi token ada kata periksa terus bocor juga sudah periksa dan betul tetap saja tiap mau isi token selalu ada kata periksa pakai adu di pln mobile lambat lebih cepat dulu pakai wa langsung ada solusi bingung sendiri deh.					
					
adu	di	aplikasi	selesai	tapi	di
lapang	enggak	ada	jadi	bingung	mau
isi	token	ada	kata	periksa	terus
bocor	juga	sudah	periksa	dan	betul
tetap	saja	tiap	mau	isi	token
selalu	ada	kata	periksa	pakai	adu
di	pln	mobile	lambat	lebih	cepat
dulu	pakai	wa	langsung	ada	solusi
bingung	sendiri	deh			

Gambar 5.15 Hasil proses *tokenizing*

f) *Filtering*

Tahap *filtering* merupakan tahap dilakukannya pemilihan kata pada ulasan atau pengurangan dimensi kata di dalam corpus yang disebut *stopword*. Pada penelitian ini, daftar *stopword* yang digunakan adalah yang telah disusun oleh Fadilah Z. Tala yang berisi sebanyak 758 kata. *Stopword removal* merupakan tahap untuk menghilangkan kata-kata yang tidak berpengaruh atau tidak informatif namun seringkali muncul dalam ulasan. Kata-kata tersebut seperti kata penghubung, kata ganti orang, kata seruan, dan kata lainnya yang tidak begitu memiliki arti dalam penentuan kelas sentimen suatu ulasan. Kumpulan *stopword* akan disimpan di dalam *stoplist* dan contoh isi *stoplist* antara lain “aku”, “kamu”, “dan”, “ke”, “itu”, “selain”, “adalah”, dan masih banyak lagi. Implementasi proses *filtering* dapat dilihat pada Tabel 5.8 berikut

Tabel 5.8 Hasil proses *filtering*

Data input	Data output
kecewa padam di saat ingin buka puasa dan ketika imsak baru nyala di apk ini juga tidak ada respon di telepon hanya buat lapor semoga saya dan keluarga selalu di beri sehat dan sabar	kecewa padam buka puasa imsak baru nyala apk respon telepon buat lapor semoga keluarga beri sehat sabar

5.2.3 Hasil Proses Labelisasi Data

Pada umumnya, analisis sentimen digunakan untuk melakukan klasifikasi (pelabelan) dokumen teks ke dalam tiga kelas sentimen, yaitu sentimen positif, negatif dan netral. Cara menentukan kelas sentimen adalah dengan menghitung skor jumlah kata positif dikurangi skor jumlah kata negatif dalam setiap kalimat ulasan. Tapi pada penelitian ini akan digunakan dua pelabelan kelas sentimen, yaitu sentimen positif dan sentimen negatif. Hal ini dilakukan karena kelas sentimen

netral dianggap kurang memberikan manfaat bagi pihak PT PLN (Persero). Adapun perhitungan skor sentimen ulasan dapat dilakukan dengan persamaan (5.1) berikut.

$$\text{skor} = \text{jumlah kata positif} - \text{jumlah kata negatif} \quad (5.1)$$

Klasifikasi yang akan digunakan pada penelitian ini menggunakan data dengan sentimen positif dan negatif. Suatu ulasan diklasifikasikan sebagai sentimen positif bila mengandung pernyataan positif seperti pujian, ungkapan terima kasih, atau testimoni positif tentang aplikasi PLN Mobile. Suatu ulasan diklasifikasikan sebagai sentimen negatif bila mengandung pernyataan-pernyataan negatif seperti ketidakpuasan, penghinaan, laporan kegagalan layanan, dan sebagainya. Simulasi perhitungan skor sentimen ulasan dapat dilihat pada Tabel 5.9 berikut.

Tabel 5.9 Simulasi perhitungan skor sentimen

Ulasan	Jumlah Kata Positif	Jumlah Kata Negatif	Skor Sentimen	Label Kelas
aplikasi error menu langgan pilih titik koordinat lokasi muncul tampil pilih provinsi jawa timur muncul tolong aplikasi mudah	4	2	2	Positif
proses bayar bayar aplikasi kecewa aplikasi batal rubah batal aplikasi icon batal sekali kesan kayak paksa coba	2	5	-3	Negatif

Kalimat yang memiliki skor ≥ 0 akan diklasifikasikan ke dalam kelas positif, sedangkan kalimat yang memiliki skor < 0 diklasifikasikan ke dalam kelas negatif. Adapun hasil proses labelisasi kelas sentimen diperoleh jumlah data seperti pada Tabel 5.10 berikut.

Tabel 5.10 Distribusi kelas sentimen hasil labelisasi data

Kelas Sentimen	Jumlah Data	Persentase Data
Positif	1518	72%
Negatif	589	28%
Total	2107	100%

Berdasarkan Tabel 5.10 diketahui bahwa *dataset* hasil proses labelisasi mengalami ketidakseimbangan kelas (*imbalanced class*). Hal ini perlu untuk dihindari karena dapat berpengaruh terhadap performa atau akurasi proses klasifikasi data. Untuk itu, pada penelitian ini digunakan teknik *oversampling* yang mana akan menyeimbangkan *dataset* dengan meningkatkan ukuran sampel pada kelas minoritas. Adapun teknik *oversampling* yang diterapkan dalam sistem dilakukan dengan bantuan *library* ROSE (*Random Over-Sampling Examples*) versi 0.0-3. Setelah dilakukan proses *oversampling*, diperoleh hasil distribusi kelas sentimen seperti pada Tabel 5.11 berikut.

Tabel 5.11 Hasil proses *oversampling*

Kelas Sentimen	Jumlah Data	Persentase Data
Positif	1518	51%
Negatif	1455	49%
Total	2973	100%

5.2.4 Hasil Proses Klasifikasi Sentimen

Data latih digunakan oleh algoritma *Naïve Bayes Classifier* untuk membentuk sebuah model pengklasifikasi. Model ini merupakan representasi pengetahuan yang akan digunakan untuk prediksi kelas data baru yang belum pernah diketahui. Semakin besar data latih yang digunakan, maka akan semakin baik sistem dalam memahami pola data. Sedangkan data uji merupakan *dataset* yang digunakan untuk mengetahui tingkat akurasi dari model yang dihasilkan. Data uji digunakan untuk mengukur sejauh mana algoritma *Naïve Bayes Classifier* berhasil melakukan klasifikasi dengan benar. Data yang digunakan untuk data latih dan data uji adalah data yang telah memiliki label kelas. Adapun pada penelitian ini

dilakukan lima kali percobaan dengan perbandingan antara data latih dan data uji yang berbeda, antara lain terlihat pada Tabel 5.12 sampai Tabel 5.16 berikut.

- a) Perbandingan data latih 70% dan data uji 30%

Tabel 5.12 Data latih 70% dan data uji 30%

Kelas Sentimen	Jumlah Data	Data Latih	Data Uji
Positif	1518	1063	455
Negatif	1455	1019	436
Total	2973	2082	891

- b) Perbandingan data latih 75% dan data uji 25%

Tabel 5.13 Data latih 75% dan data uji 25%

Kelas Sentimen	Jumlah Data	Data Latih	Data Uji
Positif	1518	1139	379
Negatif	1455	1092	363
Total	2973	2231	742

- c) Perbandingan data latih 80% dan data uji 20%

Tabel 5.14 Data latih 80% dan data uji 20%

Kelas Sentimen	Jumlah Data	Data Latih	Data Uji
Positif	1518	1215	303
Negatif	1455	1164	291
Total	2973	2379	594

- d) Perbandingan data latih 85% dan data uji 15%

Tabel 5.15 Data latih 85% dan data uji 15%

Kelas Sentimen	Jumlah Data	Data Latih	Data Uji
Positif	1518	1291	227
Negatif	1455	1237	218
Total	2973	2528	445

- e) Perbandingan data latih 90% dan data uji 10%

Tabel 5.16 Data latih 90% dan data uji 10%

Kelas Sentimen	Jumlah Data	Data Latih	Data Uji
Positif	1518	1367	151
Negatif	1455	1310	145
Total	2973	2677	296

Simulasi penerapan algoritma *Naïve Bayes Classifier* dijelaskan melalui uraian berikut ini. Peneliti ingin menganalisis sentimen terhadap ulasan-ulasan pada aplikasi PLN Mobile di Google Play Store. Dalam menganalisis sentimen, data ulasan yang telah dikumpulkan selanjutnya dikategorikan pada setiap kelas sentimen apakah kata tersebut bernilai positif atau negatif. Kemudian hasil labelisasi tersebut akan dibuat menjadi sebuah data latih. Adapun data latih tersebut dapat dilihat Tabel 5.17 di bawah ini.

Tabel 5.17 Data latih untuk simulasi

No.	Ulasan	Label Kelas
1.	bantu mudah ganggu langgan murah	Positif
2.	ganggu ribet langgan ganggu mahal murah	Negatif
3.	murah bantu langgan mudah	Positif
4.	ribet mahal mudah ganggu eror	Negatif

Dari data latih tersebut, kemudian dipilih kata-kata unik dari semua ulasan tersebut sehingga menjadi dokumen matriks pada Tabel 5.18.

Tabel 5.18 Dokumen matriks untuk simulasi

	bantu	mudah	ganggu	langgan	ribet	mahal	murah	eror
Ulasan1	1	1	1	1			1	1
Ulasan2			2	1	1	1	1	
Ulasan3	1	1		1			1	
Ulasan4		1	1		1	1		1

Dari matriks tersebut kemudian dihitung dengan menggunakan persamaan (2.4) sehingga setiap kemunculan kata dalam ulasan diubah menjadi nilai TF-IDF. Berikut adalah contoh perhitungan TF-IDF untuk kata “bantu” pada ulasan 1 dengan jumlah ulasan = 2, $TF(\text{“bantu”}) = 1$, dan $IDF(\text{“bantu”}) = 1$.

$$TF-IDF(\text{ulasan1, “bantu”}) = 1 \times \log\left(\frac{3}{1}\right) = 0,477$$

Apabila proses perhitungan di atas dilakukan untuk semua dokumen dan semua kata, maka akan dihasilkan matriks seperti Tabel 5.19 berikut.

Tabel 5.19 Hasil perhitungan TF-IDF untuk dokumen matriks

	bantu	mudah	ganggu	langgan	ribet	mahal	murah	eror
Ulasan1	0,477	0,477	0,477	0,477			0,477	
Ulasan2			0,176	0,477	0,477	0,477	0,477	
Ulasan3	0,477	0,477		0,477			0,477	
Ulasan4		0,477	0,477		0,477	0,477		0,477

Kemudian hasil perhitungan TF-IDF ini akan diolah dengan menggunakan algoritma *Naïve Bayes Classifier* sehingga menghasilkan model probabilitas. Misalkan ulasan 1 dan 3 merupakan ulasan dengan kelas positif sedangkan ulasan 2 merupakan ulasan dengan kelas negatif. Sehingga matriks TF-IDF tersebut akan berubah menjadi seperti pada Tabel 5.20.

Tabel 5.20 Hasil perhitungan TF-IDF untuk dokumen matriks

	bantu	mudah	ganggu	langgan	ribet	mahal	murah	eror	label
U1	0,477	0,477	0,477	0,477			0,477		Pos
U2			0,176	0,477	0,477	0,477	0,477		Neg
U3	0,477	0,477		0,477			0,477		Pos
U4		0,477	0,477		0,477	0,477		0,477	Neg

Berdasarkan Tabel 5.20 diketahui kelas positif memiliki dua data ulasan dengan jumlah kata sebanyak 6 kata dari 8 kosakata, sedangkan kelas negatif memiliki dua data ulasan dengan jumlah kata sebanyak 6 kata dari 8 kosakata. Dari data tersebut, maka proses membangun probabilitasnya adalah sebagai berikut.

1. Kelas Positif

Tahap ini dilakukan dengan mengumpulkan seluruh kosakata yang muncul pada semua dokumen, kemudian dihitung nilai probabilitas dari setiap kosakata. Berikut adalah contoh perhitungan nilai probabilitas kata “bantu” pada kelas positif.

$$P(\text{kata}|\text{Kelas}) = \frac{N_i+1}{N_k+N_d} \quad (5.2)$$

$$P(\text{bantu}|\text{Positif}) = \frac{2+1}{6+8} = \frac{3}{14} = 0,214$$

Keterangan:

N_i = Jumlah kata “bantu” pada kelas positif adalah 2 buah

N_k = Jumlah kata pada ulasan positif adalah 6 buah

N_d = Jumlah total kosakata adalah 8 buah

Berdasarkan perhitungan tersebut, diperoleh nilai probabilitas kata “bantu” pada kelas positif adalah sebesar 0,214. Hasil perhitungan terhadap kata-kata lain pada kelas positif disajikan pada Tabel 5.21.

Tabel 5.21 Hasil perhitungan probabilitas kosakata kelas positif

Kosakata	Nilai Probabilitas
bantu	0,214
mudah	0,214
ganggu	0,142
langgan	0,214
ribet	0,071
mahal	0,071
murah	0,214
eror	0,071

2. Kelas Negatif

Dengan menggunakan teknik yang sama seperti perhitungan pada kelas positif di atas, maka hasil dari perhitungan probabilitas kata-kata kelas negatif disajikan dalam Tabel 5.22 berikut.

Tabel 5.22 Hasil perhitungan probabilitas kosakata kelas negatif

Kosakata	Nilai Probabilitas
bantu	0,071
mudah	0,142
ganggu	0,214
langgan	0,142
ribet	0,214
mahal	0,214
murah	0,142
eror	0,142

Dengan demikian, data perhitungan di atas membentuk model probabilitas yang dihasilkan oleh algoritma *Naïve Bayes Classifier* pada Tabel 5.23.

Tabel 5.23 Model probabilitas tiap kelas sentimen

Kosakata	Prob. Positif	Prob. Negatif	Klasifikasi
bantu	0,214	0,071	Positif
mudah	0,214	0,142	Positif
ganggu	0,142	0,214	Negatif
langgan	0,214	0,142	Positif
ribet	0,071	0,214	Negatif
mahal	0,071	0,214	Negatif
murah	0,214	0,142	Positif
eror	0,071	0,142	Negatif

Setelah model probabilitas yang ada pada Tabel 5.23 terbentuk, kemudian model tersebut disimpan dalam *database* sistem pengklasifikasi. Setelah itu, model tersebut akan diuji akurasi dengan menggunakan data baru yang belum diketahui kelasnya. Sebagai contoh, akan menguji data ulasan baru sebagai data uji yang isinya “eror ganggu bantu mahal” dan diperoleh hasil berupa prediksi kelas sentimen seperti terlihat pada Tabel 5.24 berikut.

Tabel 5.24 Prediksi sentimen pada data uji

Kelas	eror	ganggu	bantu	mahal	Probabilitas
Positif (P=0,5)	0,071	0,142	0,214	0,071	0,0000765
Negatif (P=0,5)	0,142	0,214	0,071	0,214	0,0002308

Pada Tabel 5.24 terdapat kolom kelas dengan masing-masing $P = 0,5$. Nilai $P = 0,5$ merupakan peluang masing-masing kelas sebelum dilakukan proses prediksi. Sementara itu, pada kolom nilai probabilitas merupakan nilai probabilitas ulasan tersebut terhadap masing-masing kelas di mana nilai yang terbesar dari data tersebut ialah hasil prediksinya. Dengan memperhatikan Tabel 5.24, maka dapat disimpulkan bahwa ulasan “eror ganggu bantu mahal” merupakan ulasan yang termasuk ke dalam kelas negatif karena nilai probabilitas kelas negatif lebih tinggi dibandingkan dengan nilai probabilitas kelas positif.

Setelah dilakukan simulasi klasifikasi sentimen pada data latih dan data uji, selanjutnya dilakukan analisis terhadap hasil klasifikasi dari algoritma *Naïve Bayes Classifier*. Analisis hasil klasifikasi *Naïve Bayes Classifier* dilakukan dengan membuat *confusion matrix* agar dapat diketahui nilai akurasi, *recall*, dan presisi. *Confusion matrix* merupakan salah satu informasi penting yang digunakan pada *machine learning* untuk mengetahui performa sebuah sistem pengklasifikasi. Pada penelitian ini, *confusion matrix* dibuat dengan menggunakan aplikasi RStudio. Berdasarkan pembagian data latih dan data uji yang berbeda, didapatkan hasil *confusion matrix* dengan nilai akurasi, presisi, *recall*, dan *F1 score* yang berbeda pula yang dapat dilihat pada Tabel 5.25 berikut.

Tabel 5.25 Hasil *confusion matrix*

Pembagian	Akurasi	Presisi	Recall	F1 Score
Data latih 70% dan data uji 30%	88,55%	91,47%	84,29%	87,74%
Data latih 75% dan data uji 25%	89,08%	90,62%	86,18%	88,51%
Data latih 80% dan data uji 20%	90,07%	91,07%	88,23%	89,63%
Data latih 85% dan data uji 15%	89,89%	90,18%	88,94%	89,55%
Data latih 90% dan data uji 10%	90,88%	92,51%	89,26%	90,85%

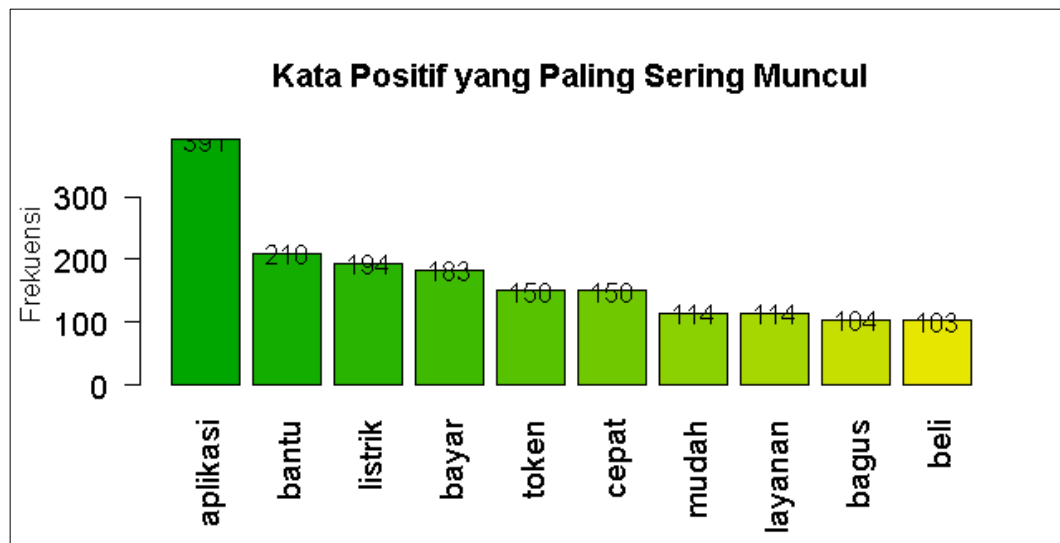
Berdasarkan hasil akurasi pada Tabel 5.25 diketahui bahwa persentase akurasi klasifikasi dengan algoritma *Naïve Bayes Classifier* akan cenderung naik seiring dengan berkurangnya jumlah data yang digunakan dalam pengujian (data uji). Adapun hasil maksimal diperoleh pada perbandingan data latih 90% dan data uji 10%, dengan akurasi tertinggi yaitu 90,88% dan presisi lebih dari 90% untuk kelas positif maupun negatif. Adapun secara keseluruhan, nilai akurasi, presisi, *recall*, maupun *F1 score* mengalami peningkatan seiring berkurangnya jumlah data uji yang digunakan dalam sistem klasifikasi sentimen.

5.2.5 Hasil Proses Asosiasi Teks

Dalam penelitian ini, dilakukan visualisasi pada semua kelas data yakni sentimen negatif maupun positif. Hal ini bertujuan untuk mengekstraksi informasi secara keseluruhan tentang topik atau bahasan yang sering diulas para pengguna aplikasi PLN Mobile. Selain itu, juga dilakukan pencarian korelasi antarkata pada masing-masing kelas data yang paling sering muncul secara bersamaan guna memperoleh informasi penting dan berguna bagi pihak yang membutuhkan.

1. Kelas Positif

Data ulasan positif yang digunakan adalah data hasil proses labelisasi yang dilakukan menggunakan kamus *lexicon* berbahasa Indonesia. Ekstraksi informasi pada ulasan positif dilakukan secara berulang-ulang hingga mendapatkan informasi tentang ulasan positif pengguna aplikasi PLN Mobile yang paling sering dibicarakan. Ulasan positif tersebut diidentifikasi berdasarkan frekuensi kata dalam ulasan, Gambar 5.16 berikut adalah visualisasi hasil ekstraksi informasi yang didapatkan dari ulasan dengan klasifikasi ulasan positif.



Gambar 5.16 Kata yang paling banyak muncul pada kelas positif

Berdasarkan hasil klasifikasi ulasan positif, dari jumlah ulasan positif sebanyak 1518 ulasan, diperoleh beberapa kata yang paling banyak muncul diantaranya adalah kata “aplikasi” dengan frekuensi sebanyak 391 kali, “bantu” sebanyak 210 kali, “listrik” 398 kali, dan seterusnya. Kata-kata yang muncul seperti pada Gambar 5.16 merupakan kata yang memiliki sentimen positif dan merupakan topik pembicaraan yang paling banyak diulas oleh pengguna aplikasi PLN Mobile. Kata-kata tersebut selanjutnya digunakan sebagai dasar untuk menemukan asosiasi dengan kata lainnya, sehingga dapat diperoleh informasi yang lebih baik. Kumpulan kata-kata yang sering muncul tersebut juga dapat ditampilkan dalam bentuk *wordcloud* seperti terlihat pada Gambar 5.17.



Gambar 5.17 Wordcloud ulasan positif

Berdasarkan visualisasi *wordcloud* dapat dilihat dengan lebih jelas topik dan kata-kata positif yang sering digunakan pengguna aplikasi PLN Mobile dalam memberikan ulasan. Semakin besar ukuran kata pada *wordcloud* menggambarkan semakin tinggi pula frekuensi kata tersebut, dan artinya semakin sering pengunjung menggunakan kata tersebut sebagai topik pembicaraan atau penilaian positif dalam ulasan. Selanjutnya, dilakukan pencarian asosiasi antarkata yang sering muncul secara bersamaan dan diperoleh hasil pada Tabel 5.26 sebagai berikut.

Tabel 5.26 Asosiasi kata pada kelas sentimen positif

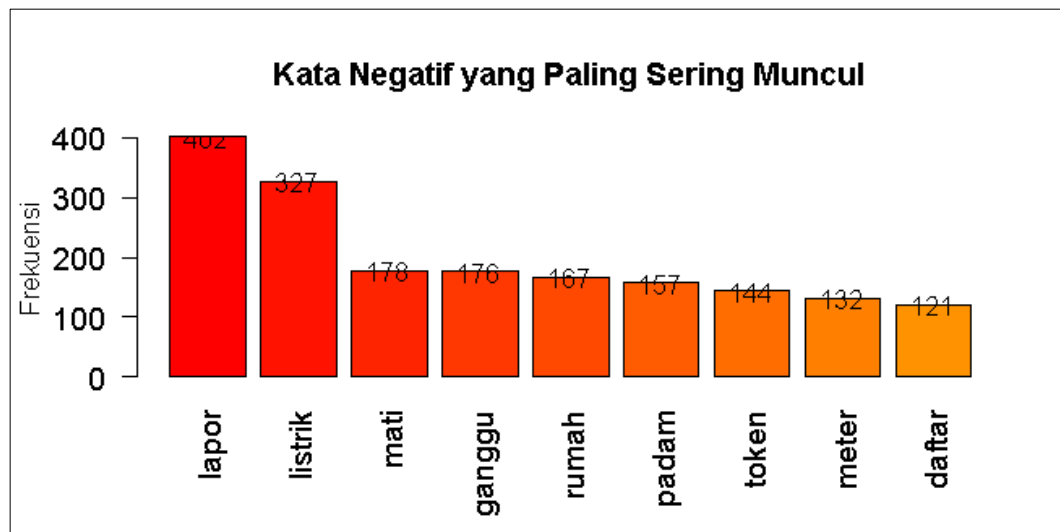
aplikasi	transaksi	layanan	cepat	harga
bagus 0,18	manual 0,43	gunung 0,18	respon 0,27	percaya 0,58
mutasi 0,18	putar 0,43	posko 0,18	tanggap 0,19	tingkat 0,58
tolong 0,18	hasil 0,22	ajak 0,17	<i>rating</i> 0,17	stabil 0,36
<i>upgrade</i> 0,17	status 0,22	bicara 0,17	ramah 0,16	admin 0,29
rekening 0,17	transfer 0,21	ekstra 0,17	posisi 0,16	<i>mobile</i> 0,29
	beli 0,20	jasa 0,17		
	tombol 0,17			
	riwayat 0,17			
	tampil 0,17			

Berdasarkan Tabel 5.26, diperoleh beberapa asosiasi kata pada klasifikasi kelas positif. Proses ekstraksi informasi dengan asosiasi dilakukan secara berulang-ulang dengan cara menyaring kata-kata yang memiliki hubungan dengan kata lain dan didasarkan pada relevansi kata pada topik yang diulas. Jika dilihat asosiasi kata yang berkaitan dengan kata “aplikasi”, dapat diperoleh informasi tentang aplikasi yang bagus, adanya fitur mutasi, tolong tambahkan fitur baru, *upgrade* atau *update* aplikasi, dan fitur rekening. Kata-kata yang berasosiasi dengan kata “transaksi” juga memberikan informasi mengenai perlunya fitur transaksi manual, transaksi yang tidak memutar atau membingungkan, adanya hasil dan status setelah melakukan transaksi, adanya fitur transaksi melalui transfer sehingga memudahkan pembelian, tombol pada aplikasi, adanya fitur riwayat transaksi, serta tampilan atau UI (*User Interface*) aplikasi.

Kata-kata yang berasosiasi dengan kata “layanan” memberikan informasi bahwa layanan PLN sudah beroperasi hingga di dataran tinggi atau pegunungan, adanya posko apabila pelanggan memerlukan bantuan, adanya ajakan dari para pengguna untuk memakai aplikasi PLN Mobile, adanya layanan *customer service* agar pelanggan dapat membicarakan keluhan, adanya layanan ekstra, dan aplikasi PLN sangat berjasa atau berkesan bagi pengguna. Kata-kata yang berasosiasi dengan kata “cepat” antara lain respon yang tanggap hingga pengguna memberikan rating yang baik, pelayanan yang ramah, dan adanya fitur *tracking* untuk mengetahui posisi pengguna yang mengalami gangguan pada layanan PLN. Terakhir dapat dilihat bahwa kata “harga” memberikan informasi bahwa biaya layanan PLN sangat terpercaya dan sesuai dengan tingkat kebutuhan pengguna, diharapkan juga agar harga selalu stabil, serta tidak adanya biaya admin pada aplikasi PLN sangat membantu dan meringankan bagi para pelanggan.

2. Kelas Negatif

Ekstraksi informasi pada ulasan negatif dilakukan secara berulang-ulang hingga mendapatkan informasi tentang ulasan negatif pengguna aplikasi PLN Mobile yang paling sering dibicarakan. Berdasarkan hasil labelisasi, ulasan negatif pengguna terhadap aplikasi sedikit lebih banyak bila dibandingkan dengan jumlah ulasan positif. Hal tersebut menunjukkan bahwa mayoritas pengguna aplikasi PLN Mobile menginginkan adanya peningkatan layanan ke arah yang lebih baik. Hasil ekstraksi informasi berupa ulasan negatif diidentifikasi berdasarkan frekuensi kata dalam ulasan. Selain itu, ekstraksi juga didasarkan pada relevansi kata dengan topik yang mengacu pada sentimen negatif. Gambar 5.18 berikut adalah visualisasi hasil ekstraksi informasi yang didapatkan dari ulasan dengan klasifikasi ulasan negatif.



Berdasarkan hasil klasifikasi ulasan negatif, diperoleh beberapa kata yang paling banyak muncul dengan topik dan dianggap relevan sebagai sentimen negatif di antaranya adalah kata “lapor” dengan frekuensi sebanyak 402 kali, “listrik” sebanyak 327 kali, “mati” sebanyak 178 kali, “ganggu” sebanyak 176 kali, “rumah” sebanyak 167 kali, dan seterusnya. Kata-kata yang muncul seperti pada Gambar 5.18 merupakan kata yang memiliki sentimen negatif berbahasa Indonesia dan merupakan topik pembicaraan yang paling banyak diulas. Kata-kata tersebut selanjutnya digunakan sebagai dasar untuk menemukan asosiasi dengan kata lainnya, sehingga dapat diperoleh informasi berupa sentimen negatif yang lebih akurat. Kumpulan kata-kata yang sering muncul tersebut dapat ditampilkan dalam bentuk *wordcloud* seperti terlihat pada Gambar 5.19.

Visualisasi wordcloud pada Gambar 5.19 memberikan gambaran yang lebih jelas tentang topik dan kata-kata negatif yang sering digunakan pengunjung dalam memberikan ulasan. Beberapa topik yang sering dibahas pengunjung di antaranya adalah tentang listrik, padam, ganggu, rumah, dan sebagainya. Selanjutnya dilakukan pencarian asosiasi antar kata yang sering muncul secara bersamaan dan diperoleh hasil pada Tabel 5.27 sebagai berikut

Tabel 5.27 Asosiasi kata pada kelas sentimen negatif

mati		ganggu		padam		susah		error	
lampu	0,53	lapor	0,35	konstruksi	0,60	jarak	0,31	server	0,38
siang	0,38	force	0,33	lokal	0,60	naik	0,31	cek	0,23
bayar	0,30	sering	0,33	pohon	0,60	nyala	0,31	string	0,23
hidup	0,30	close	0,28	rencana	0,60	saklar	0,31	subtype	0,23
hujan	0,30	mobile	0,26	roboh	0,60	setrika	0,31	sibuk	0,19
ampas	0,29	listrik	0,25	sengat	0,60				
bosan	0,29	maaf	0,20						

Tabel 5.27 menunjukkan asosiasi antarkata pada ulasan negatif, kata-kata tersebut merupakan topik yang paling sering dibicarakan pengguna dalam ulasannya. Berdasarkan tabel tersebut dapat diperoleh beberapa informasi berikut. Kata-kata yang berasosiasi dengan kata “mati” pada ulasan negatif memberikan informasi tentang lampu mati atau listrik padam, sering terjadi pemadaman pada waktu siang dan hujan, pelanggan merasa kecewa karena mereka sudah membayar, serta mereka juga meminta untuk segera hidup agar tidak ampas dan bosan.

Kata-kata yang berasosiasi dengan kata “ganggu” pada ulasan negatif memberikan informasi tentang banyaknya laporan listrik yang terganggu, aplikasi PLN Mobile yang sering *force close*, dan pihak PLN hanya sering minta maaf saat terjadi permasalahan tersebut. Kata-kata yang berasosiasi dengan kata “padam” pada ulasan negatif memberikan informasi tentang pemadaman di wilayah konstruksi, pemadaman listrik lokal, listrik yang padam saat ada pohon roboh, rencana pemadaman listrik yang kurang tersosialisasikan, dan banyaknya

masyarakat yang tersengat listrik karena kurangnya edukasi dan sosialisasi mengenai pengelolaan listrik yang bijak untuk rumah tangga.

Kata-kata yang berasosiasi dengan kata “susah” memberikan informasi tentang jarak rumah yang cukup jauh dengan teknisi PLN sehingga pelanggan susah memperoleh bantuan jika terjadi masalah, susahnya pelanggan untuk meminta naik daya listrik, listrik yang padam susah (lama) untuk nyala kembali, susah dalam pemasangan saklar, dan susah untuk menyetrika karena listrik padam. Kata-kata yang berasosiasi dengan kata “*error*” memberikan informasi tentang keluhan pelanggan mengenai server aplikasi PLN Mobile yang sering sibuk, banyak pelanggan yang meminta tim IT PLN untuk segera cek *error* tersebut, dan masalah *string subtype error* yang terjadi pada kolom input aplikasi PLN Mobile.

BAB VI PENUTUP

6.1 Simpulan

Berdasarkan pengujian yang telah dilakukan pada analisis sentimen data ulasan pengguna aplikasi PLN Mobile menggunakan *Naïve Bayes Classifier*, maka simpulan yang didapatkan oleh penulis adalah sebagai berikut.

- 1) Algoritma *Naïve Bayes Classifier* mampu dalam mengklasifikasikan data ulasan aplikasi PLN Mobile menjadi kelas sentimen positif dan negatif dengan baik setelah dilakukan pembersihan, normalisasi, dan labelisasi.
- 2) Pengujian sistem dilakukan menggunakan lima jenis perbandingan data latih dan data uji. Dengan menggunakan perbandingan data latih dan data uji sebesar 90%:10%, diperoleh tingkat akurasi tertinggi yaitu 90,88%. Artinya dari 296 data ulasan yang diujikan, terdapat 269 ulasan yang benar diklasifikasikan oleh sistem.
- 3) Berdasarkan hasil asosiasi teks yang dilakukan, secara umum dapat diketahui bahwa pengguna aplikasi PLN Mobile mayoritas membicarakan mengenai aplikasi, listrik, rumah, dan token karena selalu muncul baik pada kelas sentimen positif maupun negatif. Secara umum, metode asosiasi teks yang digunakan menunjukkan hasil ekstraksi informasi pada kelas positif di antaranya terkait aplikasi, transaksi, layanan, cepat, dan harga. Sedangkan pada kelas negatif yang sering dikeluhkan meliputi mati, ganggu, padam, susah, dan *error*.

6.2 Saran

Adapun saran yang dapat diberikan oleh penulis setelah melakukan penelitian ini terkait perbaikan dan pengembangan untuk penelitian selanjutnya adalah sebagai berikut.

- 1) Bagi pihak PT PLN (Persero), hasil ekstraksi informasi dari ulasan-ulasan yang telah diberikan oleh pengguna, khususnya ulasan yang berbentuk negatif dapat dijadikan bahan evaluasi dalam peningkatan kepuasan pengguna, memberikan pelayanan yang lebih baik untuk masa mendatang, serta dapat digunakan untuk pengembangan fitur dan pembaruan aplikasi pada versi selanjutnya.
- 2) Sistem labelisasi kelas sentimen yang digunakan dalam penelitian ini hanya terbatas pada pendeteksian sentimen tiap kata menggunakan kamus *lexicon*, sehingga kata-kata dalam bentuk frasa dan klausa belum dapat teridentifikasi dengan baik. Untuk penelitian selanjutnya, sebaiknya dapat menggunakan teknik labelisasi data yang mampu mendeteksi sentimen hingga dalam bentuk frasa dan klausa.
- 3) Dalam penelitian ini, data ulasan yang digunakan masih dibatasi untuk ulasan yang berbahasa Indonesia saja, sehingga pada penelitian selanjutnya perlu dikembangkan menggunakan ulasan pada bahasa asing ataupun bahasa daerah.
- 4) Bagi peneliti selanjutnya, dapat menggunakan pendekatan atau algoritma *machine learning* yang lain sebagai perbandingan terhadap performa algoritma *Naïve Bayes Classifier* dalam mengklasifikasikan sentimen ulasan pengguna aplikasi PLN Mobile.

DAFTAR PUSTAKA

- Abdullah, D. dan Fadlisyah, F. (2014), *Statistika Terapannya Pada Bidang Informatika*, Yogyakarta: Graha Ilmu.
- Bookhamer, P. dan Zhang, Z. J. (2016), *Knowledge Management In A Global Context: A Case Study*, Information Resources Management Journal, 29(1), 57-74.
- Chohan, S., Nugroho, A., Aji, A. M. B. dan Gata, W. (2020), *Analisis Sentimen Pengguna Aplikasi Duolingo Menggunakan Metode Naïve Bayes dan Synthetic Minority Over Sampling Technique*, Paradigma - Jurnal Komputer dan Informatika, 22(2), 139-144.
- Fanani, F. (2017), *Klasifikasi Review Software Pada Google Play Menggunakan Pendekatan Analisis Sentimen*, Skripsi, S.Kom., Universitas Gadjah Mada, Yogyakarta.
- Feldman, R. dan Sanger, J. (2007), *The Text Mining Handbook: Advanced Approaches In Analyzing Unstructured Data*, New York: Cambridge University Press.
- Gunawan, F., Fauzi, M.A. dan Adikara, P.P. (2017), *Analisis Sentimen Pada Ulasan Aplikasi Mobile Menggunakan Naive Bayes dan Normalisasi Kata Berbasis Levenshtein Distance (Studi Kasus Aplikasi BCA Mobile)*, Systemic Information System and Informatics Journal, 3(2), 1-6.
- Josi, A. dan Abdillah, L.A. (2014), *Penerapan Teknik Web Scraping Pada Mesin Pencari Artikel Ilmiah*, Jurnal Sistem Informasi, 5(2), 259-264.
- Krouska, A., Troussas, C. dan Virvou, M. (2016), *The Effect Of Preprocessing Techniques On Twitter Sentimen Analysis*, 7th International Conference on Information, Intelligence, Systems & Applications (IISA), 2016, 1–5.
- Larose, D.T. (2005), *Discovering Knowledge In Data*, New Jersey: John Wiley & Sons.
- Liu, B. (2010), *Sentiment Analysis And Subjectivity*, Handbook of Natural Language Processing, ed. 2, 627-666.
- Liu, B. (2012), *Sentiment Analysis And Opinion Mining*, Synthesis Lectures on Human Language Technologies, 5(1), 1-167.
- Manning, C.D., Raghavan, P. dan Schütze, H. (2008), *An Introduction to Information Retrieval*, New York: Cambridge University.
- Marziah, K. (2020), *What is Google Play*, (<https://www.lifewire.com/what-is-google-play-1616720>), Diakses 24 April 2021.

- Miner, G., Elder IV, J., Fast, A., Hill, T., Nisbet, R. dan Delen, D. (2012), *Practical Text Mining and Statistical Analysis For Non-Structured Text Data Applications*, Cambridge: Academic Press.
- Pamungkas, A. (2016), *Algoritma K-Means Clustering dan Naive Bayes Untuk Pengenalan Pola Tekstur*, (<https://pemrogramanmatlab.com/2016/06/24/algoritma-k-means-clustering-dan-naive-bayes-classifier-untuk-pengenalan-pola-tesktur>), Diakses 24 April 2021.
- Pang, B. dan Lee, L. (2008), *Opinion Mining and Sentiment Analysis*, Foundations and Trends in Information Retrieval, 2(1-2), 1-135.
- Pattekari, S.A.; Parveen, A. (2012), *Prediction System For Heart Disease Using Naïve Bayes*, International Journal of Advanced Computer and Mathematical Sciences, 9(3), 3028-3035.
- Prasetyo, E. (2013), *Data Mining: Konsep dan Aplikasi Menggunakan MATLAB*, Journal of Chemical Information and Modeling, 53(9), 1689-1699.
- Putranti, N.D. dan Winarko, E. (2014), *Analisis Sentimen Twitter Untuk Teks Berbahasa Indonesia Dengan Maximum Entropy dan Support Vector Machine*, IJCCS (Indonesian Journal of Computing and Cybernetics Systems), 8(1), 91-100.
- Saraswati, N.W.S. (2013), *Naïve Bayes Classifier dan Support Vector Machines Untuk Sentiment Analysis*, Seminar Nasional Sistem Informasi Indonesia SESINDO, 2013, 585-591.
- Sari, A.E., Widowati, S. dan Lhaksana, K.M. (2019), *Klasifikasi Ulasan Pengguna Aplikasi Mandiri Online di Google Play Store Dengan Menggunakan Metode Information Gain dan Naive Bayes Classifier*, eProceedings of Engineering, 6(2), .
- Shah, C. dan Jivani, A.G. (2013), *Comparison of Data Mining Classification Algorithms For Breast Cancer Prediction*, 4th International Conference on Computing, Communications, and Networking Technologies (ICCCNT), 2013, 1-4.
- Simanjuntak, R.A. (2018), *Analisis Sentimen Pada Layanan Gojek Indonesia Menggunakan Multinomial Naive Bayes*, Skripsi, S.Kom., Universitas Sumatera Utara.
- Siraj, F. dan Abdoulha, M.A. (2007), *Mining Enrolment Data Using Predictive and Descriptive Approaches*, Knowledge-Oriented Applications in Data Mining, 53–72.
- Suprpto, F. (2015), *Pengembangan Aplikasi Sentimen Analysis Menggunakan Metode Naïve Bayes (Studi Kasus Sentimen Analysis Dari Media Twitter)*, Seminar Nasional Sistem Informasi Indonesia (SESINDO), 2015, 2-3.
- Tan, Pang-Ning, Steinbach, M., Adeyeye Oshin, M., Kumar, V. dan Vipin (2006),

Introduction To Data Mining, New York: Pearson Addison Wesley.

Turland, M. (2010), *Php|architect's Guide To Web Scraping With PHP*. php[architect]. (online), (<https://www.phparch.com/books/phparchitects-guide-to-web-scraping-with-php>), Diakses 24 April 2021.

Ulwan, M.N. (2016), *Pattern Recognition Pada Unstructured Data Teks Menggunakan Support Vector Machine dan Association*, Skripsi, S.Kom., Program Studi Statistika Universitas Islam Indonesia.

Zafikri, A. (2008), *Implementasi Vector Space Model Metode Term Frequency Inverse Document Frequency (TF-IDF) Pada Sistem Temu Kembali Informasi*, Skripsi, S.Kom., Universitas Udayana.