TESIS – KS142501

PERBAIKAN KINERJA PRAPROSES KARAKTER BERULANG DALAM MENGENALI KATA PADA KLASIFIKASI SENTIMEN BERBAHASA INDONESIA

FACHRIAN ANUGERAH
5215201009

DOSEN PEMBIMBING
Prof. Ir. ARIF DJUNAIDY, M.Sc., Ph.D.
195810051986031003

PROGRAM MAGISTER
JURUSAN SISTEM INFORMASI
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2017

Halaman ini sengaja dikosongkan

THESIS – KS142501

IMPROVING THE PERFORMANCE OF REPEATED CHARACTERS PREPROCESSING IN RECOGNIZING WORDS IN THE INDONESIAN SENTIMENT CLASSIFICATION

FACHRIAN ANUGERAH
5215201009

SUPERVISOR
Prof. Ir. ARIF DJUNAIDY, M.Sc., Ph.D.
195810051986031003

MAGISTER PROGRAM
DEPARTMENT OF INFORMATION SYSTEMS
FACULTY OF INFORMATION TECHNOLOGY
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2017

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Komputer (M.Kom)
di
Institut Teknologi Sepuluh Nopember
oleh
Fachrian Anugerah
NRP. 5215201009

Tanggal Ujian : 13 Juli 2017
Periode Wisuda : September 2017

Disetujui oleh :

Prof. Ir. Arif Djunaidy, M.Sc., Ph.D.
NIP. 195810051986031003 **(Pembimbing)**

Dr. Ir. Aris Tjahyanto, M.Kom.
NIP. 196503101991021001 **(Penguji)**

Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D.
NIP. 198201202005012001 **(Penguji)**

Dekan
Fakultas Teknologi Informasi

Dr. Agus Zainal Arifin, S.Kom., M.Kom.
NIP. 197208091995121

Halaman ini sengaja dikosongkan

PERBAIKAN KINERJA PENGKLASIFIKASI SENTIMEN BERBAHASA INDONESIA MELALUI PRAPROSES PENGHAPUSAN KARAKTER BERULANG

Nama Mahasiswa : Fachrian Anugerah
NRP : 5215201009
Pembimbing : Prof. Ir. Arif Djunaidy, M.Sc., Ph.D.

ABSTRAK

Data yang relevan didapatkan melalui tahap praproses dengan menghilangkan noise agar data yang akan diolah sesuai dengan kebutuhan. Penghilangan noise tersebut dilakukan dengan menghapus karakter berulang, karena karakter ini sering dijumpai pada data twitter akibat kesalahan penulisan. Permasalahan akan muncul ketika memproses kata yang berulang, sehingga menyebabkan kata akan kehilangan makna dan tidak dapat diproses dengan baik. Penelitian ini bertujuan untuk melakukan modifikasi penghapusan karakter berulang dengan menambahkan pengukuran similarity dan mengukur tingkat kesamaan dengan kamus.

Ada empat jenis pengulangan (kata baku mengandung pengulangan yang mengalami kesalahan pengulangan karakter lebih dari satu jenis, mengandung pengulangan yang tidak mengalami kesalahan pengulangan karakter, tidak mengandung pengulangan yang mengalami kesalahan pengulangan karakter, dan tidak mengandung pengulangan yang mengalami kesalahan pengulangan karakter lebih dari satu jenis) yang akan diselesaikan menggunakan modifikasi penghapusan karakter untuk meningkatkan kualitas hasil analisis sentiment menggunakan (SVM). Penelitian ini menggunakan tiga cara pengujian yaitu membandingkan tanpa, dengan, dan modifikasi penghapusan karakter berulang.

Hasil pengujian menunjukkan bahwa modifikasi yang dilakukan menunjukan performa klasifikasi paling baik dengan nilai akurasi sebesar 74.46%, sedangkan dengan metode *illicker* menghasilkan nilai 71.71%, dan dengan metode *jaccard* menghasilkan nilai 68.04%. Modifikasi yang dilakukan memiliki peran yang signifikan dari aspek kesalahan makna dari kata, hasil terbaik dari modifikasi penghapusan karakter dengan kata dikenali sebesar 59%. Selain itu modifikasi yang dilakukan dapat meningkatkan kinerja pada tahap *stemming* dan *stop words*. Peningkatan kinerja *stemming* dibuktikan dengan jumlah kata yang dapat dikenali sebesar 682 kata. Di sisi lain peningkatan kinerja *stop words* dibuktikan dengan terdapat 86 kata yang dapat direduksi sehingga dapat menurunkan tingkat keberagaman kata yang memiliki arti dan maksud yang sama.

Kata kunci : penghapusan karakter berulang, similarity, sentimen, jenis pengulangan karakter, SVM

Halaman ini sengaja dikosongka

IMPROVING THE PERFORMANCE OF REPEATED CHARACTERS PREPROCESSING IN RECOGNIZING WORDS IN THE INDONESIAN SENTIMENT CLASSIFICATION

By	: Fachrian Anugerah
Student Identity Number	: 5215201009
Supervisor	: Prof. Ir. Arif Djunaidy, M.Sc., Ph.D.

ABSTRACT

Relevant data is obtained through the pre-process by removing the noise so that the data to be processed in accordance with the needs. Noise removal is done by deleting repetitive characters, as the characters are often encountered in twitter data due to errors. This study aims to analyze the relevant results of the pre-process removal of repeated characters in the Indonesian sentiment classification. This is obtained by modifying the removal of characters repeatedly to calculate the similarity to determine the level of similarity with the dictionary.

There are four types of characters repetitions were analyzed using repetitive character removal modifications to improve the quality of sentiment results using Support Vector Machines (SVM). Three ways of testing are done to analyze the deletion of repetitive characters by comparing: without, with, and modification of repetitive character removal.

The test results show that the modifications performed show the best classification performance with an accuracy of 74.46%, whereas with Illecker method produces a value of 71.71%, and Jaccard method produces a value of 68.04%. The modification performed has a significant role in the aspect of the meaning of the word, the best result of the character removal modification with a recognizable word of 59%. In addition, modifications made to improve performance at stemming and stop words. Improved stemming performance is evidenced by the number of words that can be recognized for 682 words. On the other hand improvement in performance of stop words is evidenced by 86 words that can be reduced so as to decrease the level of diversity of words that have the same meaning

Keywords: removing repeated characters, sentiment, classification, support vector machines

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas berkat rahmat dan ridho-Nya sehingga penulis dapat menyelesaikan tesis dengan judul "PERBAIKAN KINERJA PRAPROSES KARAKTER BERULANG DALAM MENGENALI KATA PADA KLASIFIKASI SENTIMEN BERBAHASA INDONESIA". Penyusunan tesis ini dibuat sebagai salah satu syarat kelulusan program magister jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya.

Penulis menyadari selama menempuh pendidikan dan proses penyelesaian tesis ini penulis memperoleh bantuan dan dukungan dari berbagai pihak. Dalam kesempatan kali ini, penulis mengucapkan terima kasih yang sebesar-besarnya kepada pihak-pihak yang membantu pengerjaan tesis ini, antara lain:

1. Kedua orang tua, kakak, adik dan keluarga yang telah memberikan doa, motivasi serta dukungan kepada penulis.
2. Bapak Prof. Ir. Arif Djunaidy, M.Sc., Ph.D. yang telah sabar dan telaten membimbing serta membagikan ilmu dan waktunya kepada penulis dalam pengerjaan tesis ini.
3. Bapak Dr. Ir. Aris Tjahyanto, M.Kom. dan Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D. yang telah memberikan banyak kritik dan saran untuk perbaikan penelitian ini.
4. Seluruh Bapak dan Ibu dosen serta karyawan di program magister jurusan Sistem Informasi ITS yang telah membagikan ilmu dan inspirasi kepada penulis.
5. Rekan-rekan keluarga besar program magister Sistem Informasi ITS angkatan 2015 yang telah memberikan bantuan dan dukungan kepada penulis selama mengikuti perkuliahan dan proses penelitian ini berlangsung.
6. Teman-teman dan pihak lain yang tidak dapat penulis cantumkan namanya satu per satu yang telah mendoakan, memberikan bantuan, dukungan serta sumbangan pemikiran dalam proses penyelesaian tesis ini.

Semoga Allah SWT senantiasa memberikan berkat dan anugerah-Nya serta membalas semua kebaikan yang telah dilakukan. Penulis menyadari banyak kekurangan yang terdapat dalam penelitian ini, oleh karena itu kritik dan saran yang bersifat membangun akan selalu diterima oleh penulis. Semoga penelitian ini dapat memberikan manfaat dan wawasan yang berguna bagi pengembangan ilmu pengetahuan dan bagi pembaca.

Surabaya, 13 Juli 2017

Penulis

DAFTAR ISI

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Halaman ini sengaja dikosongkan

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1. Latar Belakang

Twitter merupakan salah satu layanan jejaring sosial dimana penggunanya dapat mengirim dan membaca pesan kesesama pengguna yang sering disebut dengan *tweet*. Twitter tidak hanya digunakan sebagai media sosial untuk berbagi informasi pribadi, namun juga digunakan untuk mendapatkan informasi terbaru mengenai produk yang dicari oleh penggunanya yang disebut pengikut (*follower*). Di sisi lain, perusahaan pemilik produk dapat memanfaatkan twitter sebagai layanan pelanggan (*customer services*). Pada akhir tahun 2012 sebanyak 87.5% perusahaan berencana untuk menggunakan media sosial untuk layanan pelanggan. 71,2% mendapatkan tanggapan positif dengan digunakannya media sosial untuk layanan pelanggan (Falcon Design Studio, 2012).

Berdasarkan data yang dihimpun *We Are Social*, terdapat kenaikan pengguna internet di Indonesia selama satu tahun, dimulai dari Januari 2015 sampai Januari 2016 yakni sekitar 15%. Berdasarkan data dari Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), sampai saat ini pengguna internet di Indonesia telah mencapai 88.1 juta pengguna dengan 48% diantaranya merupakan pengguna internet harian (Damar, 2016). Dari jumlah pengguna internet di Indonesia, terdapat 79 juta pengguna aktif dengan jumlah pengguna yang mengakses melalui perangkat *mobile* sebesar 66 juta orang dengan penggunaan aplikasi yang mendominasi adalah aplikasi *chatting* dan media sosial (Damar, 2016). Seiring dengan meningkatnya penggunaan internet, banyak pengguna yang memanfaatkan internet untuk saling menghubungkan satu sama lain dalam ruang *cyber* dan menunjukkan sentimen mereka dalam bentuk komentar di situs jaringan sosial yang berbeda seperti Twitter (Khan *et al*., 2016). Dengan lebih dari 600 juta pengguna twitter, rata-rata sekitar 500 juta *tweet* perhari (internetlivestat.com, 2017).

Penggalian opini adalah proses yang digunakan untuk menganalisa percakapan pada suatu peristiwa, topik, atau produk (Amarouche *et al*., 2015). Pada dasarnya, penggalian opini digunakan untuk menggali "bagaimana orang berpikir

tentang hal tertentu, orang atau ide" kemudian dilakukan pengolahan untuk dapat memberikan informasi yang berguna dalam mengambil keputusan sesuai dengan sentimen individu (Khan *et al*., 2016). Penggalian opini memiliki tujuan untuk mengklasifikasikan komentar menjadi opini positif atau negatif (Amarouche *et al*., 2015) dan menentukan emosi dari sebuah dokumen (Arifiyanti, 2015). Penggalian opini tidak hanya berguna untuk klien, tetapi juga membantu organisasi untuk mengevaluasi pendapat dan perilaku klien terhadap perusahaan dan produk mereka. Organisasi bisa mendapatkan ulasan tentang produknya langsung dari klien melalui jaringan sosial seperti twitter (Basari *et al*., 2013). Tahapan pada penggalian opini dimulai dengan pengumpulan data. Data didapatkan dengan mudah melalui sumber seperti blog, media sosial dan berita web berisi opini atau pendapat. Kemudian dilakukan praproses pada opini atau pendapat untuk mendapatkan data yang sesuai dengan penelitian dan dilanjutkan dengan melakukan ekstrasi fitur dan pemilihan fitur, dan diakhiri dengan proses klasifikasi (Amarouche *et al*., 2015). Setiap tahapan pada praproses bergantung pada tahap yang dilakukan sebelumnya. Hasil dari tahap sebelumnya akan bertindak sebagai masukan ke tahap berikutnya. Jika proses pada tahap tertentu tidak berjalan dengan benar akan mempengaruhi hasil pada proses tahap selanjutnya (Shirbhate *and* Deshmukh, 2016).

Penelitian ini membahas mengenai bagaiman cara mendapatkan hasil yang relevan dari tahap praproses sebelum dilakukannya klasifikasi pada data twitter yang memiliki keunikan tersendiri sehingga membutuhkan pengolahan yang berbeda. Tantangan dalam penggunaan data twitter memiliki keterbatasan jumlah kata pada tiap *twett* yaitu 140 karakter setiap *tweet*, penggunaan bahasa informal (Bouazizi *and* Ohtsuki, 2015), dan penulisan karakter yang berulang (Arifiyanti, 2015). Penggunaan bahasa informal sering tidak sesuai dengan EYD, tata bahasa yang buruk, dan sarkasme atau tidak terdaftar dalam kamus sehingga membutuhkan praproses yang lebih (Bahrainian *and* Dengel, 2013). Selain itu dokumen dalam bahasa Indonesia mempunyai keunikan tersendiri, karena kata-kata dalam bahasa Indonesia dapat berubah bentuk saat mendapatkan imbuhan. Oleh karena itu perlu dilakukan modifikasi pada teknik *stemmer*, sehingga dapat mengembalikan kata ke dalam bentuk dasarnya pada teks bahasa Indonesia (Tala, 2003). Untuk itu perlu dilakukan beberapa tahapan ekstrasi fitur pada praproses untuk memproses kalimat

opini yang memiliki struktur yang tidak baku agar dapat diproses dan menghasilkan nilai akurasi yang tinggi.

Banyak penelitian sebelumnya yang mekalukan pendekatan analisis sentimen mendapatkan kinerja klasifikasi yang lebih rendah dengan menggunakan teks *tweet* dibanding ketika diterapkan pada teks yang lebih panjang (Bouazizi *and* Ohtsuki, 2015). Kinerja klasifikasi yang tinggi tidak hanya dipengaruhi oleh algoritma pengklasifikasi, namun terdapat faktor lain yang dapat mempengaruhi kinerja klasifikasi. Salah satunya adalah pemilihan ekstrasi fitur yang digunakan dalam praproses. Penggunaan algoritma dan ekstrasi fitur memiliki pengaruh pada tinggi rendahnya nilai akurasi karena keduanya saling melengkapi, kinerja algoritma bergantung pada penggunaan algoritma sedangkan ekstrasi membantu membuang derau dan data yang tidak relevan (Coutinho *and* Figueiredo, 2013). Namun tidak semua ekstrasi fitur harus digunakan, karena tidak semua ekstrasi fitur dibutuhkan dan relevan dengan data twitter, sehingga pemilihan fitur yang digunakan juga menentukan nilai akurasi yang dihasilkan dari klasifikasi. Jika sumber adalah situs media sosial, penggunaan bahasa dan konversi tertentu perlu ditangani (Bhuta *and* Doshi, 2014).

Pada penelitian yang dilakukan Khan, ia menggunakan *tokenizer* dan *pos tagging* dalam praproses dalam klasifikasi komentar pada Youtube (Khan *et al*., 2016). Pada penelitian lainnya dilakukan *pos tagging* dengan menggunakan *pos tagger* Bahasa Indonesia yang dikembangkan oleh Pisceldo (Vidya *et al*., 2015). Berbagai macam kendala ketika mengidentifikasi sentimen pada twitter salah satunya dikarena keterbatasan karakter pada *tweet*, orang sering menggunakan bentuk singkatan yang bisa membawa arti yang berbeda. Penggunaan bahasa gaul dan kalimat dengan tata bahasa yang tidak baku membuat ditambahnya tahapan pada proses praproses (Gokulakrishnan *et al*., 2012). Penelitian yang dilakukan oleh Arifiyanti, ingin menguji metode ekstraksi fitur dalam praproses data teks konten twitter berbahasa Indonesia dengan cara melakukan beberapa modifikasi pada *case folding* dengan tidak menghapus beberapa simbol karena simbol tersebut digunakan untuk mendeskripsikan *emoticon*, memodifikasi *stemmer*, menambahkan konversi *emoticon*, dan konversi kata tidak baku dengan menambahkan daftar kata tidak baku (Arifiyanti, 2015). Pada penelitian tersebut, didapatkan nilai akurasi dari hasil

pengujian tanpa menggunakan ekstrasi fitur sebesar 91,65% dan dengan menggunakan ekstrasi fitur sebesar 94,67% yang artinya terdapat peningkatan akurasi sebesar 3,02% ketika menggunakan seluruh tahap ekstrasi fitur. Kekurangan pada penelitian tersebut diantaranya adalah tidak optimalnya pada tahap *stemming*. Hal itu disebabkan karena terdapat tahapan sebelum dilakukan *stemming* yaitu penghapusan karakter berulang. Penghapusan karakter berulang ini menghapus karakter berulang baik kata baku maupun kata tidak baku seperti "hingga" menjadi "hinga" yang menyebabkan kata tersebut tidak dapat diproses dengan baik pada tahap berikutnya. Begitu pula dengan penelitian yang dilakukan oleh (Amolik et al., 2016; Arifiyanti, 2015; Illecker, 2015; Shirbhate and Deshmukh, 2016) mereka juga melakukan tahapan penghapusan karakter berulang dengan menghilangkan karakter yang mengalami pengulangan. Namun permasalahan terjadi ketika memproses kata yang memang memiliki perulangan pada kata bakunya seperti "sehingga", "tunggu", dan "saat". Tahap penghapusan karakter berulang akan menghapus perulangan "sehingga" menjadi "sehinga", "tunggu" menjadi "tungu", dan "saat" menjadi "sat" sehingga kata akan kehilangan maknanya dan tidak dapat diproses dengan baik pada tahap berikutnya. Berbeda dengan penelitian yang dilakukan (Choi et al., 2014), ia melakukan perhitungan *smilarity* pada kamus untuk memperbaiki kesalahan penulisan perulangan dengan cara menemukan kata yang memiliki kesamaan terdekat

Dari data twitter yang didapatkan pada tanggal 23 Januari 2017 hingga 17 Februari 2017 sebanyak 5840 tweet, terdapat 8022 kata memiliki perulangan karakter dengan 65% diantaranya adalah perulangan karakter yang tidak dapat dikenali kata baku. Pada penelitian yang dilakukan (Garg, 2014), ia berpendapat bahwa kesalahan yang terjadi dikarenakan kesalahan perulangan lebih sering terjadi jika dibandingkan penggunaan emotikon. Dari hasil penelitiannya menunjukkan penghapusan karakter berulang dapat mereduksi lebih banyak jumlah kata dibanding penggunaak emotikon.

Setelah tahapan ekstrasi fitur selesai kemudian dilanjutkan proses klasifikasi. Ada beberapa algoritma yang dapat digunakan dalam proses klasifikasi teks, diantaranya adalah *Naive Bayes Classifier* (NBC), *Support Vector Model* (SVM), *Logistic Regression*, *Maksimum Entropy* (ME), *Multinominal Random Forest*,

Decision Tree (DT), dan *K-Nearest Neighbor* (KNN). Dari beberapa algoritma klasifikasi tersebut banyak penelitian yang telah membuktikan keunggulan yang dimiliki SVM dengan memiliki nilai akurasi yang tinggi jika dibandingkan dengan algoritma yang lain. Pada penelitian yang dilakukan Vidya, dilakukannya pengujian klasifikasi sentimen dengan menggunakan SVM, NB, dan DT. Hasil penelitiannya menunjukan kinerja SVM lebih unggul dari NB dan DT (Vidya *et al*., 2015). SVM banyak diterapkan dalam konteks klasifikasi terutama klasifikasi dengan sumber data berupa teks dan telah dibuktikan oleh banyak penelitian (Arifiyanti, 2015). Meskipun data yang didapatkan dari sosial media seperti twitter memiliki karakteristik yang unik, SVM dapat mencapai akurasi yang tinggi untuk mngklasifikasikan sentimen saat menggabungkan fitur yang berbeda (Akaichi, 2013). Dalam penelitian ini proses kalasifikasi menggunakan algoritma SVM dalam melakukan proses klasifikasi.

1.2. Rumusan Masalah

Untuk mendapatkan data yang relevan dengan penelitian, perlu dilakukan tahapan praproses dengan menghilangkan derau agar data yang akan diolah sesuai dengan kebutuhan penelitian. Mereduksi data sangat penting dilakukan dalam klasifikasi teks. Data yang tidak relevan dan berlebihan sering menurunkan kinerja klasifikasi algoritma baik dalam kecepatan dan akurasi klasifikasi dan juga kecenderungan untuk mengurangi *overfitting* (Aurangzeb et al., 2010). Salah satu dari tahapan pada praproses adalah tahap penghapusan karakter berulang. Penghapusan karakter berulang perlu dilakukan karena pada data twitter sering dijumpai penggunaan karakter berulang yang disebabkan karena kesalahan penulisan maupun kesengajaan pengguna seperti “iyaaaa”, “kapaan”.

Pada penelitian yang dilakukan oleh (Illecker, 2015), ia melakukan penghapusan karakter berulang dengan cara menghapus karakter yang terdapat perulangan didalamnya seperti “apaaa” menjadi “apa”, “heloooooow” menjadi “helow”. Begitu pula dengan penelitian yang dilakukan oleh (Amolik *et al*., 2016; Arifiyanti, 2015; Shirbhate *and* Deshmukh, 2016) mereka juga melakukan tahapan penghapusan karakter berulang dengan menghilangkan karakter yang mengalami pengulangan. Namun permasalahan terjadi ketika memproses kata yang memang

memiliki perulangan pada kata bakunya seperti “sehingga”, “tunggu”, dan “saat”. Tahap penghapusan karakter berulang akan menghapus perulangan “sehingga” menjadi “sehinga”, “tunggu” menjadi “tungu”, dan “saat” menjadi “sat” sehingga kata akan kehilangan maknanya dan tidak dapat diproses dengan baik pada tahap berikutnya. Selain itu permasalahan terjadi ketika kata tersebut mendapatkan imbuhan seperti kata “pelanggannya”, yang terdiri dari kata “pelanggan” kemudian mendapatkan imbuhan “nya” yang mengakibatkan kata tersebut mengalami perulangan karakter “g” dan “n” sehingga tidak dapat dikenali oleh kamus Bahasa Indonesia. Jika kata tersebut diproses menggunakan penghapusan katakter berulang, karakter “g” dan “n” akan direduksi menjadi “pelanganya” yang mengakibatkan kata tidak dapat dikenali meskipun imbuhan “nya” telah dihapus. Berbeda dengan penelitian yang dilakukan (Choi et al., 2014), ia melakukan perhitungan *smilarity* pada kamus untuk memperbaiki kesalahan penulisan perulangan dengan cara menemukan kata yang memiliki kesamaan terdekat.

Ada empat macam jenis perulangan yang terjadi pada teks berbahasa Indonesia diantaranya adalah 1) kata baku mengandung perulangan yang mengalami perulangan karakter lebih dari satu jenis karakter seperti “pelanggannya”, “mengganggu”; 2) Kata baku mengandung perulangan yang tidak mengalami perulangan karakter seperti “maaf”, “manfaat”; 3) kata baku tidak mengandung perulangan yang mengalami perulangan karakter seperti “kecewaaa”, “lagiii”; 4) kata baku tidak mengandung perulangan yang mengalami perulangan karakter lebih dari satu jenis karakter seperti “pertanyaannya”, “masiiihhh”. Dari empat jenis perulangan yang telah dijelaskan memerlukan penanganan yang berbeda untuk mendapatkan kata yang sesuai dengan kata bakunya.

. Berdasarkan permasalahan yang telah dijabarkan, maka penelitian lanjut perlu dilakukan untuk menjawab persoalan “Bagaimana melakukan modifikasi penghapusan karakter berulang agar kata yang diperbaiki tetap dapat dikenali dalam kamus Bahasa Indonesia?” dan “Apakah modifikasi penghapusan karakter berulang yang dilakukan dapat meningkatkan nilai akurasi pada klasifikasi sentiment pada twitter?”. Untuk dapat menjawab persoalan tersebut perlu dilakukan penelusuran lebih mendalam mengenai penghapusan karakter yang dilakukan pada tahapan penghapusan karakter berulang.

1.3. Tujuan

Tujuan dari penelitian ini adalah menambahkan penilaian keserupaan untuk mencari kemiripan dengan kata baku pada tahap penghapusan karakter berulang untuk meningkatkan kualitas hasil analisis sentimen.

1.4. Batasan Penelitian

Batasan penelitian dalam penelitian ini diantaranya adalah:

a. Bahasa yang digunakan pada dokumen *tweet* adalah Bahasa Indonesia.
b. Data *tweet* yang digunakan dalam penelitian ini diambil dari akun pengaduan penyedia layanan telekomunikasi seluler di Indonesia, diantaranya adalah telkomsel, indosat, xl, dan smartfren di twitter tahun 2017.
c. Analisis yang dilakukan mengenai analisis sentimen sehinga hanya memproses karakter teks dan mengabaikan karakter selain teks.
d. Fokus penelitian adalah penghapusan karakter berulang pada tahap praproses.
e. Proses penghapusan karakter mengabaikan kata yang dalam bentuk tidak baku.

1.5. Kontribusi Penelitian

Hasil dari penelitian ini diharapkan dapat memberikan kontribusi baik secara teori maupun secara praktik. Kontribusi secara teori didapatkan dari metode yang digunakan dalam memodifikasi tahap penghapusan karakter berulang sehingga hasil modifikasi penghapusan karakter berulang tersebut dapat digunakan sebagai masukan dalam berbagai jenis penelitian penggalian sentimen yang akan datang. Dengan dilakukannya modifikasi diharapkan tidak ada kata yang kehilangan makna setelah melewati tahap penghapusan karakter berulang dan kata yang mengalami perulangan karakter akan diubah menjadi kata baku sehingga akan meningkatkan akurasi dalam proses klasifikasi teks.

Kontribusi praktik dapat diterapkan bagi penyedian layanan telekomunikasi untuk melakukan analisis sentimen pengguna terhadap produk dan layanan yang mereka berikan.

1.6. Sistematika Penulisan Tesis

Sistematika penulisan dokumen laporan penelitian tesis ini dibagi menjadi lima bab yakni sebagai berikut:

BAB I PENDAHULUAN

Dalam bab ini dijelaskan mengenai latar belakang, rumusan masalah, tujuan penelitian, kontribusi penelitian, dan sistematika.

BAB II LANDASAN TEORI DAN KAJIAN PUSTAKA

Dalam bab ini dijelaskan mengenai kajian pustaka dari berbagai penelitian yang memiliki kaitan dengan penelitian ini. Kajian pustaka ini bertujuan untuk memperkuat dasar dan alasan dilakukannya penelitian ini. Selain kajian pustaka, dalam bab ini juga dijelaskan mengenai teori-teori terkait yang bersumber dari buku, jurnal, ataupun artikel yang berfungsi sebagai dasar dalam melakukan penelitian.

BAB III METODOLOGI PENELITIAN

Dalam bab ini dijelaskan mengenai langkah-langkah penelitian beserta metode yang digunakan. Langkah-langkah penelitian dijelaskan dalam sebuah diagram alur yang sistematis dan akan dijelaskan tahap demi tahap.

BAB IV UJI COBA DAN ANALISIS HASIL

Dalam bab ini akan dilakukan uji coba terhadap metode penggalian teks yang telah dirancang sebelumnya. Uji coba ini dilakukan berdasarkan skenario uji coba yang telah dirancang sebelumnya. Selain itu dalam bab ini juga dijelaskan mengenai analisis hasil uji coba.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari penelitian ini dan juga saran bagi penelitian mendatang yang berasal dari kekurangan ataupun temuan dari penelitian ini.

BAB II
KAJIAN PUSTAKA DAN LANDASAN TEORI

Dalam bab in akan dijelaskan mengenai dasar teori yang berhubungan dengan penelitian yang akan dilakukan. Selain itu akan dibahas mengenai penelitian-penelitian sebelumnya. Teori yang dijelaskan diantaranya meliputi konsep mengenai twitter; penggalian data teks; praproses yang meliputi pembersihan derau (*noise cleaning*), *tokenizing*, *case folding*, penghapusan kata henti (*stop words removal*), penghapusan karakter berulang, *stemming*, dan konversi kata tidak baku; kalsifikasi; dan pengukuran kinerja. Sedangkan penelitian-penelitian sebelumnya yang terkait diantaranya meliputi praproses pada klasifikasi sentimen pada twitter dan klasifikasi sentimen dengan menggunakan metode SVM (*Support Vector Machine*).

2.1. Penelitian Terkait

Pada data twitter memiliki tantangan tersendiri sehingga membutuhkan pengolahan yang berbeda. Hal itu disebabkan karena terdapat batasan jumlah karakter sebesar 140 untuk mengirim *tweet*. Pembatasan itu terkadang membuat pengguna tidak dapat mengekspresikan diri mereka sehingga mereka menggunakan bahasa informal, mempersingkat beberapa kata dengan menghapus vokal, seperti "*story*" menjadi "*stry*" (Keretna *et al*., 2013), dan pada penelitian lain menunjukan hasil yang lebih rendah dengan menggunakan data twitter dibanding dengan teks yang lebih panjang (Bouazizi *and* Ohtsuki, 2015), sehingga memiliki potensi terjadi kesalahan ejaan dan kalimat tidak terstruktur dengan baik. akurasinya tergantung pada memilih fitur yang relevan (Keretna *et al*., 2013).

Untuk menggali sentimen pada data twitter, perlu dilakukan praproses terlebih dahulu. Praproses dilakukan untuk menghilangkan data yang tidak relevan dengan penelitian, sebab data yang tidak relevan dan berlebihan dapat menurunkan kinerja klasifikasi (Aurangzeb *et al*., 2010). Sebelum dilakukan praproses, data yang telah terkumpul perlu dilakukan pengelompokan sesuai dengan sentimen yang dikandung. (Gokulakrishnan *et al*., 2012) melakukan pengelompokan dengan cara memberi bobot pada tiap kata yang memiliki sentimen positif dan negatif. Dengan

diberinya bobot, maka setiap dokumen dapat diketahui sentimen yang mendominasi di dalamnya. Pada tahap praproses dilakukan konversi *emoticon*, *uppercase identification*, *lower casing*, ekstrasi *URL*, konversi *username* dan *hashtag*, penghapusan tanda baca, penghapusan kata henti, penghapusan kata kunci, penghapusan karakter berulang. Kemudian dilakukan pengujian dengan menggunakan berbagai macam algoritma pengklasifikasi diantaranya adalah *Naive Bayes*, *Naïve Bayes Multinominal*, *Complement Naive Bayes*, *DM NBteks*, *Bayesian Logistic Regression*, SMO, SVM, J48, *Random Forest*, *Lazy IBK*. Hasil terbaik didapatkan algoritma SMO dengan nilai akurasi sebesar 81.86% (Gokulakrishnan *et al*., 2012).

Pada penelitain lain yang dilakukan (Arifiyanti, 2015), ia melakukan modifikasi ketika melakukan pembersihan derau dengan menghapus tanda-tanda *hashtag*, dan URL, kemudian *case folding* yang meruhbah karakter menjadi huruf kecil dan karakter selain huruf "a" – "z" dihapus. Selain itu terdapat pengecualian karakter yang tidak dihapus karena karakter tersebut digunakan dalam penulisan *emoticon*. Kemudian melakukan tahapan konversi kata tidak baku dengan mengubah token berbentuk kata tidak baku menjadi bentuk kata bakunya. Kemudian dilakukan klasifikasi dengan menggunakan algoritma SVM. Hasil penelitian menunjukan bahwa dengan menggunakan seluruh tahap ekstrasi fitur didapat akurasi sebesar 94.67% dan 91.65% ketika tidak menggunakan ekstrasi fitur (Arifiyanti, 2015). Rangkuman penelitian terkait dijelaskan pada Tabel 2.1.

Pada penelitian yang dilakukan oleh (Amolik *et al*., 2016; Arifiyanti, 2015; Shirbhate *and* Deshmukh, 2016; Garg, 2014) mereka melakukan tahapan penghapusan karakter berulang dengan menghilangkan karakter yang mengalami pengulangan. Namun penghapusan karakter yang digunakan tidak dapat memproses kata yang memang memiliki perulangan pada kata bakunya. Tahap penghapusan karakter berulang akan menghapus perulangan pada kata sehingga kata akan kehilangan maknanya dan tidak dapat diproses dengan baik pada tahap berikutnya. Oleh karena itu penelitian ini dilakukan untuk memperbaiki tahapan penghapusan karakter berulang sehingga tidak ada kata yang kehilangan makna setelah melewati tahap penghapusan karakter berulang.

Tabel 2.1 Penelitian Terkait Mengenai Klasifikasi Sentimen

No	Peneliti	Judul	Praproses	Algoritma Klasifikasi	Hasil	Temuan
1	(Arifiyanti, 2015)	Ekstrasi Fitur Pada Konten Jejaring Sosial Twitter Berbahasa Indonesia Dalam Peningkatan Kinerja Klasifikasi Sentimen	1. *Cleansing* 2. *Case folding* 3. *Tokenizing* 4. *Removing repetition* 5. Penghapusan *stop words* 6. *Emotic conversation* 7. *Stemming* 8. *Non st*and*art language conversation*	SVM	Akurasi dengan menggunakan seluruh praproses sebesar 94.67% sedangkan tanpa menggunakan praproses sebesar 91.65%	Algoritma *stemmer* yang kurang optimal, disebabkan tahapan penghapusan karakter berulang tidak optimal sehingga mengakibatkan tidak dapat diproses oleh *stemmer*.
2	(Gokulakrishnan *et al*., 2012)	*Opinion Mining and Sentimen Analysis on a Twitter Data Stream*	1. Konversi *Emoticon*s 2. *Uppercase Identification* 3. *Case folding* 4. URL *Extraction* 5. Konversi *usename* dan *hashtags* 6. Penghapusan *stop words* 7. *Compression of Words* 8. *Removing Skewness*	*Naïve Bayes, Naïve Bayes Multinominal, Complement Naïve Bayes, DM NBteks, Bayesian Logistic Regression, SMO, SVM, J48, Random Forest, Lazy IBK*	Hasil akurasi terbaik didapatkan dengan menggunakan SMO sebesar 81.86%, kemudian diikuti CNB dan DMNBteks di kisaran 80%	Perlu dilakukan percobaan dengan beberapa sampel yang berbeda agar mendapatkan hasil yang akurat. Alasan untuk memiliki beberapa samplesis ialah jika kita mengambil kesimpulan hanya berdasarkan sampel tunggal, hasil dapat menyesatkan karena mereka mencerminkan karakteristik yang spesifik. Dampak dari karakteristik sampel tertentu pada kesimpulan akhir dan akurasi yang didapatkan.
3	(Basari *et al*., 2013)	*Opinion Mining of Movie Review using*	1. Data *cleansing* 2. *Case folding*	SVM, SVM-PSO	Didadapatkan nilai akurasi sebesar	Ditemukan kejanggalan ketika menggunakan data cleansing dengan

Tabel 2.1 Penelitian Terkait Mengenai Klasifikasi Sentimen (lanjutan)

No	Peneliti	Judul	Praproses	Algoritma Klasifikasi	Hasil	Temuan
		Hybrid Method of Support Vector Machine and Particle Swarm Optimization	3. Tokenization 4. *Stemming* 5. n-gram 6. TF 7. TF-IDF		71.87% dengan menggunakan SVM dan 77% dengan menggunakan SVM-PSO	tanpa menggunakan data cleansing. Nilai akurasi lebih tinggi sebesar 0.33% didapatkan tanpa melakukan data *cleansing*. Hal itu membuktikan bahwa pada tahap normalisasi tidak berjalan dengan optimal.
4	(Haddi *et al*., 2013)	*The Role of* Teks *Pre-processing in Sentimen Analysis*	*1. Data cleaning* 2. *Tokenizing* 3. Konversi kata tidak baku 4. *Stemming* 5. *Stop word* removal 6. TF-IDF	SVM	Hasil akurasi yang didapatkan adalah 93.5%	Penggunaan metode preprosesing yang tepat dapat meningkatkan kinerja klasifikasi secara signifikan.
5	(Garg, 2014)	*Sentiment Analysis of Twitter Feeds*	*1. Remove Hashtags* *2. Remove username* *3. Remove URL* *4. Convertion emoticon* *5. Stemming* *6. Remove repeating character* *7. Stemming*	*Naive Bayes*	Akurasi terbaik didapatkan sebesar 86.68%	
6	(Shirbhate *and* Deshmukh, 2016)	*Feature Extraction for Sentiment Classification on Twitter Data*	*1. Case folding* *2. Conversion emoticon*	*Naive Bayes*	Akurasi keseluruhan didapatkan sebesar 88.4%	Setiap tahapan pada praproses bergantung pada tahap yang dilakukan sebelumnya. Keluaran dari tahap sebelumnya akan bertindak sebagai

Tabel 2.1 Penelitian Terkait Mengenai Klasifikasi Sentimen (lanjutan)

No	Peneliti	Judul	Praproses	Algoritma Klasifikasi	Hasil	Temuan
			3. *Conversion username* 4. *Remove tweets having few Words* 5. *Removing URL* 6. *Remove repeating character* 7. *POStag* 8. *Remove stop words* 9. *Stemming*			Masukan ke tahap berikutnya. Jika proses pada tahap tertentu tidak berjalan dengan benar, maka akan mempengaruhi hasil keluaran pada proses tahap selanjutnya.
7	(Amolik *et al*., 2016)	*Twitter Sentiment Analysis of Movie Reviews using Machine Learning Techniques*	1. *Case folding* 2. *Removing URL* 3. *Replace username* 4. *Tokenizing* 5. *Removing hashtag* 6. *Stop words* 7. *Removing repeating character*	*Naive Bayes, SVM*	Didapatkan akurasi sebesar 75% dengan SVM dan 65% dengan *Naive Bayes*	

2.2. Data Twitter

Twitter adalah jaringan sosial yang memungkinkan penggunanya untuk mengirim pesan teks dengan maksimal 140 karakter, yang dikenal sebagai *tweet*s. Twitter terstruktur dengan *follow* dan *follower*s, dimana setiap pengguna diberikan kebebasan untuk melakukan *follow* atau *follower*s pada akun lainnya. Ada juga kemungkinan untuk mengirim pesan pribadi ke profil lain. Hal ini juga memungkinkan mengirim video, foto dan mengarahkan pembaca ke halaman web lain melalui link. Twitter adalah jaringan sosial yang dinamis yang memungkinkan setiap pengguna untuk memiliki akses ke informasi yang terus diterbitkan. (Daniel *et al*., 2017)

Twitter menyediakan *REST API* untuk pengembang yang memungkinkan untuk mengakses data status, dan profil pengguna. Twitter juga menyediakan akses pengembang ke sejumlah informasi secara *real time* melalui *Streaming API*. *Tweet*s dapat dikelompokkan dengan kata-kata *hashtag* didahului oleh karakter "#", digunakan untuk menandai kata kunci atau topik *tweet*. Selain itu, pengguna dapat membuat *tweet* yang diposting oleh pengguna lain.

2.3. Penggalian Data

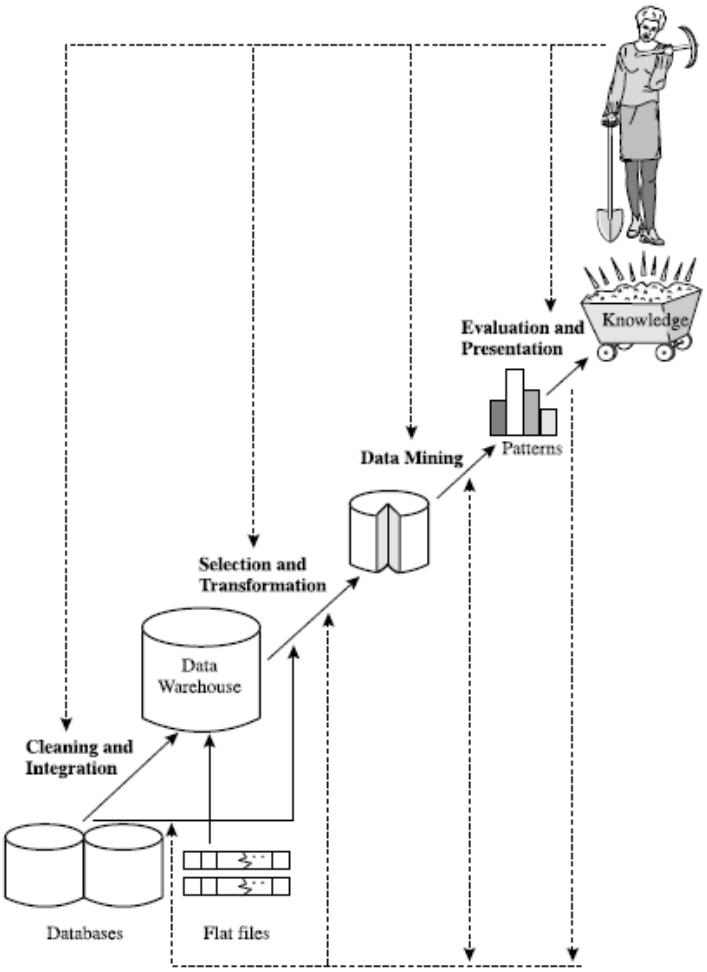
Pengertian penggalian data menurut Pramudiono dalam (Aprilla *et al*., 2013) adalah analisis otomatis dari data yang berjumlah besar atau kompleks dengan tujuan untuk menemukan pola atau kecenderungan yang penting yang biasanya tidak disadari keberadaanya. Pada penggalian data dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, yaitu :

1. Klasifikasi, suatu teknik dengan melihat pada kelakuan dan atribut dari kelompok yang telah didefinisikan. Teknik ini dapat memberikan klasifikasi pada data baru dengan memanipulasi data yang ada yang telah diklasifikasi dan dengan menggunakan hasilnya untuk memberikan sejumlah aturan.
2. Asosiasi, digunakan untuk mengenali kelakuan dari kejadian-kejadian khusus atau proses dimana hubungan asosiasi muncul pada setiap kejadian.

3. Klasitering, digunakan untuk menganalisis pengelompokkan berbeda terhadap data, mirip dengan klasifikasi, namun pengelompokkan belum didefinisikan sebelum dijalankannya tool penggalian data (Aprilla *et al*., 2013).

Ada beberapa langkah penting dalam proses penggalian data seperti yang disajikan pada Gambar 2.1. Penggalian data sebagai suatu proses terdiri dari urutan berulang dari langkah-langkah berikut: (Han *and* Kamber, 2000)

1. Pembersihan data, untuk menghilangkan kebisingan dan data yang tidak konsisten.
2. *Data inetegration*, dimana beberapa sumber data dapat digabungkan.
3. Pemilihan data, di mana data yang relevan dengan tugas analisis akan diambil dari *database*.
4. *Data transformation*, dimana data diubah atau dikonsolidasikan ke dalam bentuk yang sesuai untuk pertambangan dengan melakukan ringkasan atau operasi agregasi.
5. Penggalian data, proses penting dimana metode yang diterapkan untuk mengekstrak pola data.
6. *Pattern evaluation*, untuk mengidentifikasi pola-pola yang benar-benar menarik yang mewakili pengetahuan didasarkan pada beberapa tindakan *interestingness*.
7. *Knowledge presentation*, di mana visualisasi dan pengetahuan teknik representasi digunakan untuk menyajikan pengetahuan untuk pengguna.



Gambar 2.1 Tahapan Penggalian Data (Han *and* Kamber, 2000)

2.4. Penggalian Data Teks

Pada penelitian penggalian data lebih berfokus pada data terstruktur seperti relasional, transaksional, dan *data warehouse*. Namun, dalam kenyataannya sebagian besar dari informasi yang tersedia disimpan dalam database teks yang terdiri dari koleksi besar dokumen dari berbagai sumber, seperti artikel berita, makalah penelitian, buku, perpustakaan digital, pesan *e-mail*, dan halaman web. Data teks yang tersimpan pada *database* adalah data semi terstruktur yang artinya data benar-benar terstruktur seperti judul, penulis, tanggal publikasi, kategori, tetapi juga mengandung beberapa komponen teks sebagian besar tidak terstruktur, seperti abstrak dan isinya. Sehingga sulit untuk dilakukannya analisis dan penggalian informasi dari data tersebut. Pengguna membutuhkan alat untuk membandingkan dokumen yang berbeda, peringkat pentingnya dan relevansi dokumen, atau

menemukan pola dan tren di beberapa dokumen. Dengan demikian, pertambangan teks telah menjadi tema yang semakin populer dan penting dalam penggalian data. (Han *and* Kamber, 2000).

2.5. Praproses Data

Praproses data perlu dilakukan sebab data yang kita dapatkan terkadang tidak lengkap (hilangnya suatu atribut), berisik (mengandung kesalahan, atau nilai-nilai *outlier* yang menyimpang dari yang diharapkan), dan tidak konsisten (misalnya, mengandung perbedaan dalam kode departemen digunakan untuk mengkategorikan item). Hal itu dapat terjadi karena beberapa alasan diantaranya adalah tidak dianggap penting pada saat diproses sehingga tidak di simpan. Data yang relevan mungkin tidak tercatat karena kesalahpahaman, atau karena kerusakan peralatan. Data yang tidak konsisten dengan data lain yang tercatat mungkin telah dihapus. Rekaman sejarah atau modifikasi data mungkin telah diabaikan. Kesalahan dalam transmisi data juga dapat terjadi. Jika pembersihan data tidak dilakukan menyebabkan hasil dari setiap penggalian data tidak dapat dipercayai kebenarannya (Han *and* Kamber, 2000).

Pada umumnya tahapan praproses data terdiri:

1. *Tokenizing*, yaitu proses pemecahan dokumen menjadi beberapa token dengan menggunakan *whitespace* (“ “) sebagai pemisahnya. Tidak semua fitur yang dikembalikan oleh *tokenization* yang algoritma harus digunakan, karena daftar berisi banyak fitur yang tidak relevan. Oleh karena itu, pemilihan fitur Metode yang digunakan juga menentukan akurasi dari klasifikasi. (Bhuta *and* Doshi, 2014).
2. Pembersihan detau, digunakan untuk menghapus dokumen yang tidak relevan dengan penelitian.
3. *Case folding*, yaitu proses merubah semua teks huruf kapital menjadi huruf kecil. Selain itu karakter selain huruf (a-z) seperti angka, simbol akan dihilangkan.
4. Penghapusan kata henti, yaitu proses penghapusan token yang sering muncul sehingga tidak memiliki makna lagi (Tala, 2003).

5. *Stemming*, yaitu proses untuk merubah masing-masing token menjadi bentuk kalimat dasarnya dengan menghapus imbuhan, sisipan, dan akhiran.
6. Penghapusan karakter berulang, yaitu proses yang digunakan untuk merubah kata yang mengandung perulanga karakter menjadi karakter tunggal.

2.6. Jaro Winkler

Jaro-Winkler *distance* merupakan varian dari Jaro distance *metric* yang merupakan sebuah algoritma untuk mengukur kesamaan antara dua *string*. Menurut Cohen pada penelitiannya, ia berpendapat bahwa Jaro Winkler ditujukan untuk mengukur kesamaan *string* yang pendek (Cohen *et al*., 2003). Hasil dari perhitungan ini menghasilkan nilai 0 – 1 dimana 0 menandakan tidak ada kesamaan antar dokumen dan 1 menandakan terdapat kesamaan pada dokumen (Dreßler *and* Ngomo, 2014). Pada penelitian yang dilakukan (Sumathi *et al*., 2015), ia mencoba menguji JaroWinkler *distance* dengan Hamming *distance*, dan Dameran Levenshtein *distance* dengan nilai presisi, *recall*, dan *f-measure* lebih unggul dari Hamming *distance*, dan Dameran Levenshtein *distance* (Sumathi *et al*., 2015). Begitu pula dengan penelitian yang dilakukan oleh (Cohen et al., 2003), Jaro Winkler dapat mengungguli Levenstein, Levenstein Winkler, SoftTFIDF, dan Jaro. Algoritma Jaro Winkler yang dituliskan pada persamaan 2.1 dan 2.2. Dalam persama ini, parameter d_j, m, $|s_1|$, $|s_2|$, t, d_w, l, dan p berturut-turut menyatakan nilai Jaro *distance*, jumlah karakter yang sama, panjang string 1, panjang string 2, setengah jumlah karakter transposisi, nilai Jaro Winkler *distance*, panjang karakter yang sama sebelum ditemukan ketidaksamaan, dan konstanta *prefix weight* (*default* = 0.1).

$$d_j = \frac{1}{3} \times \left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m} \right) \quad (2.1)$$

$$d_w = d_j + \left(l.p\left(1 - d_j\right) \right) \quad (2.2)$$

2.7. *Support Vector Machine*s

*Support Vector Machine*s (SVM) merupakan metode yang menjanjikan untuk klasifikasi menggunakan data *linear* dan *non linear*. Algoritma ini bekerja dengan menggunakan pemetaan *nonlinear* untuk mengubah data training ke dimensi yang lebih tinggi. Dalam dimensi baru ini, akan mencari garis pemisah (*hyperplane)* optimal *linear* (yaitu, "batas keputusan" memisahkan tupel dari satu kelas dari yang lain). Dengan pemetaan *non linear* yang tepat ke dimensi yang cukup tinggi, data dari dua kelas selalu dapat dipisahkan dengan garis. SVM menemukan garis ini menggunakan dukungan vektor (pelatihan tupel) dan *margin* (didefinisikan oleh support vektor). SVM merupakan algoritma yang sangat akurat, karena kemampuan mereka untuk model *non linear* yang kompleks (Han *and* Kamber, 2000).

Ide dasar dari algoritma SVM adalah mencari garis yang optimal dengan nilai *margin* maksimal seperti pada Gambar 2.3. Dimulai dengan mendefinisikan persamaan suatu garis pemisah yang dituliskan pada persamaan 2.3. Dalam persama ini, parameter W, n, dan b berturut-turut menyatakan bobot vektor (W_1, W_2, …, W_n), jumlah atribut, dan *scalar*.

$$W.X + b = 0 \tag{2.3}$$

Berdasarkan pada atribut A1, A2 pada Gambar 2.2 dengan permisalan tupel pelatihan X = (x1, x2) dimana x1 dan x2 merupakan nilai dari atribut A1 dan A2, dan jika b dianggap sebagai suatu bobot tambahan w_0, maka persamaan suatu garis pemisah dapat ditulis ulang seperti pada persamaan 2.4 (Han *and* Kamber, 2000).

$$W_0 + W_1X_1 + W_2X_2 = 0 \tag{2.4}$$

Sedangkan setiap titik yang terletak di atas garis seperti pada persamaan 2.5.

$$W_0 + W_1X_1 + W_2X_2 > 0 \tag{2.5}$$

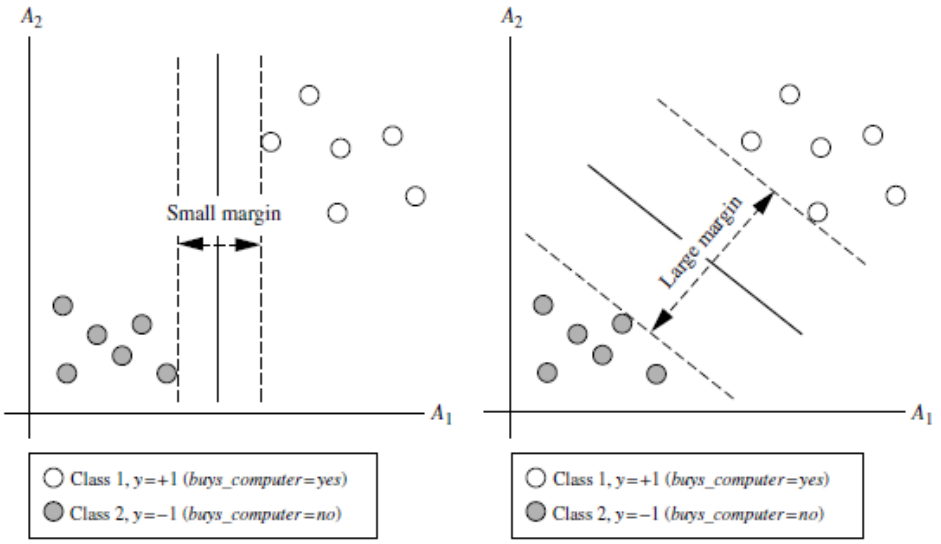
Dan sebaliknya, setiap titik yang terletak di bawah garis seperti pada persamaan 2.6.

$$W_0 + W_1X_1 + W_2X_2 < 0 \quad (2.6)$$

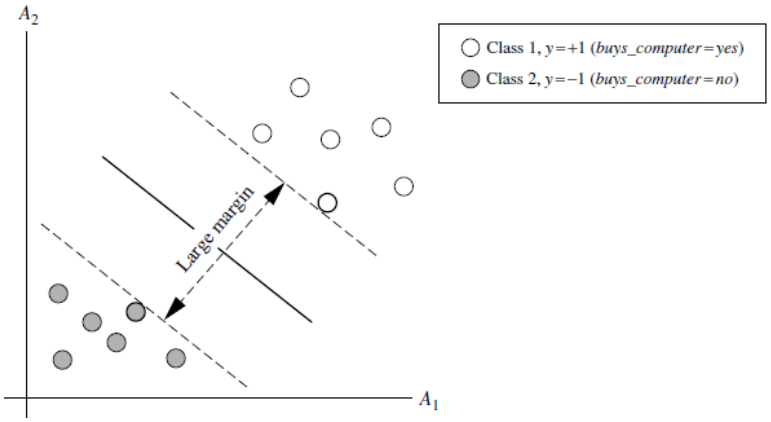
Sehingga didapatkan dua persamaan garis seperti pada persamaan 2.7 dan 2.8.

$$H_1 : w_0 + w_1x_1 + w_2x_2 \geq 1 \quad for\ y_0 = +1 \quad (2.7)$$

$$H_2 : w_0 + w_1x_1 + w_2x_2 \leq 1 \quad for\ y_0 = -1 \quad (2.8)$$



Gambar 2.2 Margin Minimum dan Maksimum (Han *and* Kamber, 2000)



Gambar 2.3 Pemisahan Dua Kelas Dengan Margin Maksimum (Han *and* Kamber, 2000)

Berikut ini adalah beberapa fungsi kernel yang umum digunakan diantaranya yaitu : (Han *and* Kamber, 2000).

1. *Linier* kernel

$$K(X_i, X_j) = \theta(X_i).\theta(X_j) \quad (2.9)$$

2. *Polynomial* kernel

$$K(X_i, X_j) = (X_i, X_j + 1)^h \quad (2.10)$$

3. *Gaussian RBF* kernel

$$K(X_i, X_j) = e^{-||X_i - X_j||^2/2\sigma^2} \quad (2.11)$$

4. *Sigmod* kernel

$$K(X_i, X_j) = \tanh(kX_i.X_j - \delta) \quad (2.12)$$

Kernel *linier* digunakan ketika data yang akan diklasifikasi dapat terpisah dengan sebuah garis. Sedangkan kernel *non linier* digunakan ketika data hanya dapat dipisahkan dengan garis lengkung atau sebuah bidang pada ruang dimensi tinggi.

2.8. Pengukuran Kinerja Klasifikasi

Terdapat banyak cara dalam pengukuran kinerja dalam klasifikasi. Pada penelitian yang dilakukan (Bouazizi *and* Ohtsuki, 2015), (Arifiyanti, 2015), dan (Haddi *et al*., 2013), pengujian dilakukan dengan menerapkan metode *cross validation*. *Cross validation* adalah sebuah teknik untuk menilai atau melakukan validasi keakuratan sebuah model yang dibangun berdasarkan dataset. Dalam penelitian ini menggunakan *K-Fold validation*, yakni membagi dataset menjadi sejumlah K partisi secara acak untuk diuji. Pengukuran kinerja didapatkan melalui perbandingan pada *Confusion matrix*, presisi, *recall, F-Measure* dan akurasi. *Confusion matrix* merupakan suatu metode yang biasanya digunakan untuk melakukan perhitungan akurasi pada konsep penggalian data. *Confusion matrix* berguna untuk menganalisis seberapa baik *classifier* anda dapat mengenali tupel dari kelas yang berbeda. *Confusion matrix* melakukan perhitungan dengan 3

keluaran, yaitu *recall*, presisi, dan akurasi seperti pada Tabel 2.2 (Han *and* Kamber, 2000).

Tabel 2.2 Tabel *Confusion Matrix*

		Predicted Class	
		True	***False***
Actual Class	***True***	TP *(True Positive)* Hasil yang benar	FN *(False Negatif)* Hasil yang hilang
	False	FP *(False Positive)* Hasil yang tidak terduga	TN *(True Negatif)* Hasil yang tidak benar

Presisi adalah presentase dokumen sebenarnya yang diambil *relevan* dengan *query*. Secara formal dapat didefnisikan seperti pada persamaan 2.16. *Recall* adalah presentase dokumen yang relevan dengan *query* dan pada kenyataannya di ambil. Secara formal dapat dedefinisikan seperti pada persamaan 2.17. Akurasi adalah nilai perbandingan antara nilai data yang diklasifikasikan secara benar dengan seluruh data seperti pada persamaan 2.18. *F-Measure* adalah nilai *harmonic* atau nilai rata-rata (*mean*) dari nilai presisi dan recal seperti pada persamaan 2.19 (Han *and* Kamber, 2000). Dalam persama ini, parameter TP, FP, FN, dan TN berturut-turut menyatakan jumlah klasifikasi yang benar dari data positif, jumlah klasifikasi yang salah dari data negatif, jumlah klasifikasi yang salah dari data positif, dan jumlah klasifikasi yang benar dari data negatif.

$$precision = \frac{TP}{TP + FP} \tag{2.16}$$

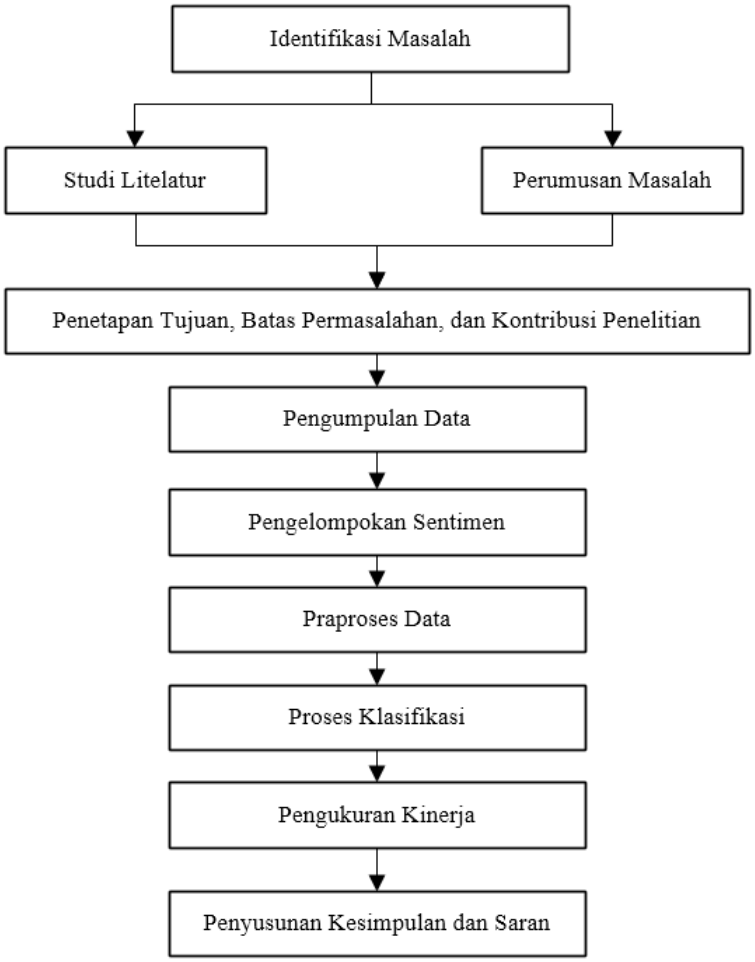
$$recall = \frac{TP}{TP + FN} \tag{2.17}$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.18}$$

$$F\ measure = \frac{2\ .precision\ .recall}{precision + recall} \tag{2.19}$$

BAB III
METODOLOGI PENELITIAN

Dalam bab ini akan dijelaskan mengenai tahapan yang akan dilakukan dalam penelitian ini seperti yang disajikan pada Gambar 3.1. Diawali dengan melakukan identifikasi masalah yang digunakan untuk menentukan ruang lingkup, perumusan masalah, penentuan tujuan, dan kontribusi penelitian. Setelah melakukan identifikasi masalah, kemudian dilanjutkan ke tahap studi litelatur. Studi litelatur dilakukan dengan mencari penelitian terdahulu dan dasar teori yang dapat menunjang dan memiliki hubungan dengan permasalahan yang diangkat dalam penelitian ini. Kemudian dilanjutkan dengan melakukan pengumpulan data, praproses data, klasifikasi, pengukuran kinerja, dan analisis hasil penelitian.



Gambar 3.1 Metode Penelitian

3.1. Pengumpulan Data

Dalam penelitian ini terdapat beberapa data yang harus disiapkan terlebih dahulu. Data tersebut terdiri dari data utama berupa komentar yang didapatkan dari *tweet* pada media sosial twitter dan data pendukung seperti data penghapusan kata henti dan data kata dasar Bahasa Indonesia. Data *tweet* yang digunakan dalam penelitian ini adalah opini pelanggan mengenai penyedia layanan telekomunikasi seluler di Indonesia pada tahun 2017, diantaranya adalah telkomsel, indosat, xl, smartfren yang memiliki layanan pengaduan di akun twitter seperti yang dijelaskan pada Tabel 3.1.

Tabel 3.1 Akun Twitter Penyedia Layanan Telekomunikasi di Indonesia

No.	Provider	Akun Twitter	Kata Kunci
1	Telkomsel	@telkomcare	%40telkomcare
2	Indosat	@indosatcare	%40indosatcare
3	Xl	@xlcare	%40xlcare
4	Smartfren	@smartfrencare	%40smartfrencare

Pengumpulan data twitter dilakukan dengan menggunakan fasilitas yang telah disediakan oleh twitter yaitu *REST API* dan aplikasi R. *REST API* menyediakan akses untuk membaca dan menulis data twitter dalam format *JSON*. Kata kunci yang digunakan dalam penelitian ini adalah akun resmi pengaduan dari masing-masing penyedia layanan telekomunikasi. Penggunaan tanda "@" (*mention*) tidak dapat digunakan ketika melakukan pencarian dengan kata kunci, karena *REST API* menggunakan kode "%40" untuk menggantikan tanda "@". Data twitter yang diambil melalui aplikasi R akan secara otomatis merubah dari format *JSON* kedalam bentuk *CSV*. Data yang diperoleh terdiri dari beberapa atribut seperti *no, text, favorited, favoriteCount, replyToSN, created, truncated, replyToSID, id, replyToUID, statusSource, screenname, retweetCount, isRetweet, retweeted, longitude*, dan *latitude*. Namun dalam penelitian ini hanya atribut *text* yang digunakan untuk melakukan klasifikasi sentimen dan *retweeted* dengan nilai "false" yang menandakan tidak ada *tweet* yang serupa. Data yang digunakan dalam penelitian ini adalah *tweet* yang mengandung perulangan didalamnya sehingga

tweet yang didak mengandung perulangan akan dibuang karena tidak relevan dengan penelitian.

Pengumpulan data dilakukan secara bertahap karena data yang didapatkan dari *REST API* tidak bersifat historis yang hanya dapat mengambil data dalam jangka waktu yang telah ditetapkan oleh twitter. Oleh karena itu proses pengumpulan data dilakukan dalam tiga minggu secara bertahap. Berikut merupakan contoh dokumen yang didapatkan menggunakan *REST API* seperti pada Tabel 3.2.

Tabel 3.2 Contoh Dokumen Keluaran REST API

No	Dokumen
1	@IndosatCare min, mau nanya dong untuk ngecek kartu SIM udah 4G atau belum itu dimana yaa?
2	@IndosatCare kenapa indosat jaringan 4g lte mentok di 1mbps, lokasi ungaran kabupaten semarang apa emang ada https://t.co/5BRixc7tc4
3	@IndosatCare selamat mlm kakak, mau tanya ni knp aku isi pls kok gak tambah masa aktifnya??
4	@IndosatCare indosat gangguan ada apa yaa
5	@IndosatCare malem min minta tolong jaringan nomer saya di perbaiki dapetnya 4G buat internet aja gak bisa sama sekali no sy dm ya
6	@IndosatCare min untuk nomor indosat ane paket 20 GB 4G n 8GB 3g mengalami limit speed di 130KBps ,min tolong solusinya
7	@IndosatCare ini nunggu berapa hari lg??
8	@IndosatCare ka ko internetnya lambat banget yaa , padahal posisi jaringan di 4G tlong solusinya ka makasih:)?

3.2. Pengelompokan Sentimen

Tahap pengelompokan sentimen digunakan untuk mengelompokan dokumen yang telah terkumpul kedalam kelas sentimen positif, sentimen netral, dan sentimen negatif sesuai dengan sentimen yang terkandung pada *tweet*. Pengelompokan ini dilakukan secara manual berdasarkan jenis opininya seperti pada Tabel 3.3.

Tabel 3.3 Contoh Pengelompokan Sentimen

Tweet	Kelompok
@TelkomCare DM sudah diterima. Terima kasih banyak admin untuk informasinya yang sangat membantu	Positif
@46dree Terima kasih atas saran dan masukan yg Bpk Andre berikan. cc:@smartfrencare Tks :) ^Yoga	Positif

Tabel 3.3 Contoh Pengelompokan Sentimen (lanjutan)

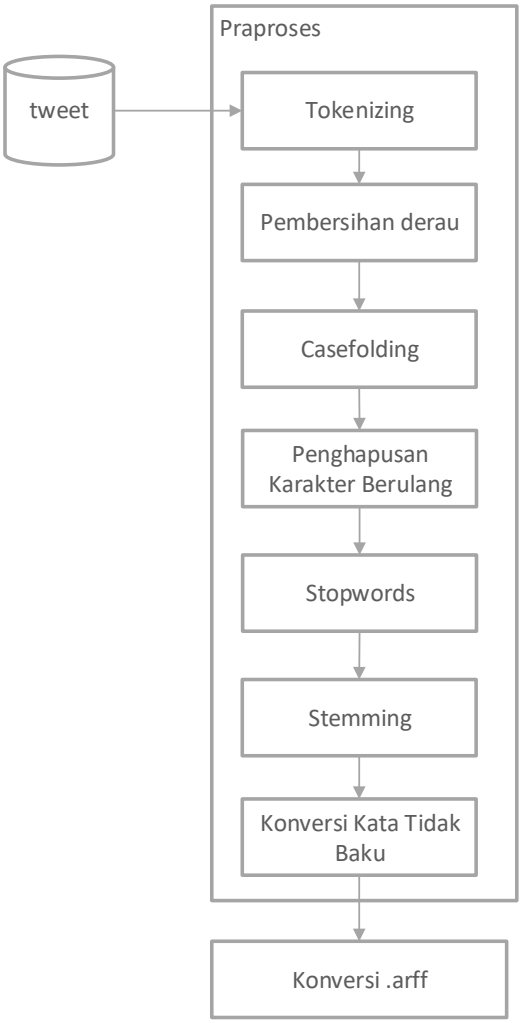
Tweet	**Kelompok**
internet mati 2 hari. dijanjikan teknisi mau datang tapi ditanya kapan datangnya ga bisa jawab. teknisi cuma 1 apa gmn ? @TelkomCare	Negatif
Ini knp ya pulsa tdnyaada 31rb trus tiba2 kepotong sendiri tinggal 29rbrb. Saya ga pake apa2 buat sms engga, tlp juga engga @IndosatCare	Negatif
@XLCare saya pakai samsung galaxy note 5	Netral
@smartfrencare cek DM min....	Netral

3.3. Praproses Data

Sebelum melakukan proses klasifikasi, data yang telah terkumpul perlu melalui tahap praproses data terlebih dahulu. Hal ini dilakukan dengan cara menghilangkan derau agar data yang akan diolah sesuai dengan kebutuhan penelitian. Mereduksi data sangat penting dilakukan dalam klasifikasi teks, karena data yang tidak relevan dan berlebihan sering menurunkan kinerja klasifikasi algoritma baik dalam kecepatan dan akurasi klasifikasi dan juga kecenderungan untuk mengurangi *overfitting* (Aurangzeb *et al*., 2010).

Tahap pertama praproses dari penelitian ini adalah *tokenizing*, yaitu tahapan yang berfungsi untuk memecah dokumen menjadi beberapa token. Setelah dokumen dipecah menjadi beberapa token, kemudian dilakukan pembersihan derau yang bertujuan untuk menghapus dokumen yang tidak relevan dalam dokumen seperti menghapus *URL*, simbol "@", "#". Setelah dibersihkan dari dokumen yang tidak relevan kemudian dilanjutkan dengan *case folding*, yaitu merubah krakter pada token menjadi huruf kecil. Hal ini dilakukan agar mengurangi keragaman karakter. Selain itu juga dilakukan penghapusan karakter selain huruf a-z. Tahap keempat dilanjutkan dengan melakukan penghapusan karakter berulang. Penghapusan karakter ini dilakukan untuk menghapus karakter-karakter yang mengalami kesalahan pada penulisannya, karena pada data twitter banyak mengandung bahasa informal, tidak sesuai dengan EYD, tata bahasa yang buruk sehingga membutuhkan praproses yang lebih (Bahrainian *and* Dengel, 2013). Tahap kelima adalah penghapusan kata henti yang digunakan untuk menghapus kata-kata yang sering muncul. Tahap enam adalah *stemming*. *Stemming* bertujuan untuk mengubah kata menjadi kata dasarnya dengan cara membuang imbuhan,

sisipan, dan akhiran. Setelah tahap *stemming* dilakukan, maka setiap token telah dirubah menjadi kata dasar dan hanya tersisa token yang relevan dengan penelitian. Tahap terahir adalah konversi kata tidak baku, konversi ini dilakukan dengan membandingkan dengan daftar kalimat tidak baku. Tahapan praproses digambarkan pada Gambar 3.2. Hasil dari praproses digunakan sebagai Masukanan pada proses klasifikasi.



Gambar 3.2 Tahapan Praproses

3.3.1. *Tokenizing*

Tahap *tokenizing* digunakan untuk memecah dokumen menjadi beberapa token dengan menggunakan *whitespace* (“ “) sebagai pemisahnya. Contoh penerapan *tokenizing* dapat dilihat pada Tabel 3.4.

Tabel 3.4 Contoh Tokenizing

No	Masukan	Keluaran
1	terima kasih indosat.. saya sekarang sangat Puas..	terima; kasih; indosat..; saya; sekarang; sangat; Puas..
2	trimakasih buat indosat.. penanganan masalah 4g lambat udh tratasi dan sangat cepat responnya	trimakasih; buat; indosat..; penanganan; masalah; 4g; lambat; udh; tratasi; dan; sangat; cepat; responnya

3.3.2. Pembersihan Derau

Tahap pembersihan derau digunakan untuk menghapus data yang tidak relevan dengan penelitian. Simbol *hashtag*, *retwett*, *URL*, dan akun. Hal ini dilakukan untuk menghapus data yang tidak bermakna dan mereduksi data yang akan diolah. Contoh penerapan pembersihan derau dapat dilihat pada Tabel 3.5.

Tabel 3.5 Contoh Pembersihan Derau

No	Masukan	Keluaran
1	@IndosatCare woi gw pake wifi kali, maen charge2 aj, maling! https://t.co/HsfTZos0eu	woi gw pake wifi kali, maen charge2 aj, maling!
2	@XLCare hp saya sudah 4g dan kartu xl nya juga 4g dengan nomor hp 087874633025 ,, tapi koneksi internet saya hanya rata rata 150kb/s, why?	hp saya sudah 4g dan kartu xl nya juga 4g dengan nomor hp 087874633025 ,, tapi koneksi internet saya hanya rata rata 150kb/s,?

3.3.3. *Case Folding*

Tahap *case folding* dilakukan untuk merubah huruf kapital menjadi huruf kecil. Selain itu karakter selain huruf (a-z) seperti angka, simbol akan dihilangkan. Contoh penerapan *case folding* dapat dilihat pada Tabel 3.6.

Tabel 3.6 Contoh Penggunaan *Case Folding*

No	Masukan	Keluaran
1	kenapa nggak bisa cek pulsa ya? Laporan "*Error* MMI *code*" sdh dari semalam loh	kenapa nggak bisa cek pulsa ya laporan *error* mmi *code* sdh dari semalam loh
2	padahal baru pukul 11:16 PM hari ini tanggal 31 Desember 2016 berlangganan. Mohon tanggapannya	padahal baru pukul pm hari ini tanggal desember berlangganan mohon tanggapannya

Tabel 3.6 Contoh Penggunaan *Case Folding* (lanjutan)

No	Masukan	Keluaran
3	4Gnya ilang lagi woi. penyakitan banget sih kesel sumpah -_-	gnya ilang lagi woi penyakitan banget sih kesel sumpah

3.3.4. Penghapusan Karakter Berulang

Tahap penghapusan karakter berulang dilakukan untuk membenarkan dokumen dari kata yang mengalami perulangan karakter yang disebabkan oleh kesalahan pada penulisan. Pada penelitian yang dilakukan oleh (Illecker, 2015), ia melakukan penghapusan karakter berulang untuk menghapus karakter yang terdapat perulangan didalamnya. Begitu pula dengan penelitian yang dilakukan oleh (Amolik *et al*., 2016; Arifiyanti, 2015; Shirbhate *and* Deshmukh, 2016) mereka juga melakukan tahapan penghapusan karakter berulang dengan menghilangkan karakter yang mengalami pengulangan. Hal itu dilakukan karena pada data twitter, banyak sekali dijumpai penulisan kata yang tidak baku (Bahrainian *and* Dengel, 2013).

Masukan dari proses ini adalah sebuah kata yang didalamnya mengalami perulangan maupun tidak mengalami perulangan. Kemudian kata tersebut diperiksa setiap karakternya apakah mengalami perulangan atau tidak. Jika terdeteksi adanya perulangan maka dilakukan proses reduksi perulangan karakter sehingga karakter yang mengalami perulangan akan direduksi menjadi karakter tunggal seperti “apaaaaa” menjadi “apa”, “cepaat” menjadi “cepat”. Hasil dari proses penghapusan karakter berulang ini dijadikan keluaran dari tahap penghapusan karakter berulang. Contoh penggunaan penghapusan karakter berulang dapat dilihat pada Tabel 3.7.

Tabel 3.7 Contoh Penggunaan Penghapusan Karakter Berulang

No	Masukan	Keluaran
1	ogaaahhh... selama orientasi smartfren masih jadi penjual hape, selama itu…	ogah. selama orientasi smartfren masih jadi penjual hape, selama itu.
2	oalaaah, ternyataa gitu. Oke oke. Makasiih banyak yaaaah	oalah, ternyata gitu. Oke oke. Makasih banyak yah
3	2 hari pak buseeettttttt	2 hari pak buset
4	untuk mengetahui format bentuk surat kuasa penggantian kartu sim indosat bisa dilihat dmn yaa?	untuk mengetahui format bentuk surat kuasa penggantian kartu sim indosat bisa dilihat dmn ya?

3.3.5. Penghapusan Kata Henti

Tahap penghapusan kata henti dilakukan untuk menghapus token yang sering muncul sehingga tidak memiliki makna lagi. Proses *stopping* dilakukan dengan menggunakan data *stoplist* yang digunakan pada peneitian yang dilakukan (Tala, 2003). Proses penghapusan dilakukan dengan cara membandingakan token dengan *stoplist*. Jika token terdaftar pada *stoplist*, maka token tersebut akan dihapus.

3.3.6. *Stemming*

Tahap *stemming* adalah proses untuk merubah masing-masing token menjadi bentuk kalimat dasarnya dengan menghapus imbuhan, sisipan, dan akhiran. Proses *stemming* membutuhkan kamus kata dasar Bahasa Indonesia yang digunakan untuk dibandingkan tiap token dengan daftar kata dasar yang ada dikamus. Jika terdapat dalam daftar kata baku maka akan diubah menjadi kata dasarnya. Stemmer yang digunakan dalam penelitian ini adalah Sastrawi *stemmer* yang merupakan sebuah *library stemmer* yang menerapkan algoritma *Nazief, Adriani, Confix Stripping*, dan *Enhanced Confix Stripping*. Dengan menggunakan algoritma tersebut, banyak persoalan *stemming* seperti mencegah *overstemming* dengan menggunakan kamus kata dasar, mencegah *understemming* dengan menggunakan aturan-aturan tambahan, dan kata bentuk jamak seperti "buku-buku" menjadi "buku" dapat diatasi oleh Sastrawi *stemmer* (Sastrawi, n.d.). Contoh penerapan *stemming* dapat dilihat pada Tabel 3.8.

Tabel 3.8 Contoh Stemming

No	Masukan	Keluaran
1	dibantu	Bantu
2	dibeli	Beli
3	mendapatkan	Dapat
4	menghambat	Hambat
5	berubah	Ubah

3.3.7. Konversi Kata Tidak Baku

Tahap konversi kata tidak baku dilakukan dengan mencocokan masing-masing token dengan daftar kata tidak baku. Jika token tersebut memiliki kesamaan

dengan yang ada didalam daftar kata tidak baku, maka token tersebut akan dirubah menjadi kalimat bakunya. Contoh penerapan konversi kata tidak baku dapat dilihat pada Tabel 3.9.

Tabel 3.9 Contoh Konversi Kata Tidak Baku

No	Masukan	Keluaran
1	ini sinyal di Royal Mediterrania Garden, 085781069761 kok ga stabil lg? Ga usa suruh gue kasi rt rw, cpt betulin	ini sinyal di Royal Mediterrania Garden, 085781069761 kok tidak stabil lagi? tidak perlu suruh saya kasi rt rw, cepat benarkan
2	oke berarti sinyal disininya saja yg lemot ya.	oke berarti sinyal disininya saja yg lambat ya.
3	di Cibiru atas,kota bandung, sinyal smartfren modem andromax lelet sekali. tdk bisa akses internet di hp dan laptop.	di Cibiru atas,kota bandung, sinyal smartfren modem andromax lambat sekali. tidak bisa akses internet di hp dan laptop.

Setelah tahap terahir dilakukan, didapatkan keluaran dari praproses yaitu dokumen-dokumen yang sesuai dengan kebutuhan dan relevan dengan penelitian. Dokumen-dokumen tersebut kemudian dikonversi menjadi *.arff* (*Attribute-Relation File Format*) untuk dilanjutkan ke proses selanjutnya yaitu kalsifikasi.

3.4. Pengukuran Kinerja Praproses

Pengukuran kinerja praproses dilakukan untuk mengetahui seberapa baik kinerja dari praproses yang digunakan untuk mereduksi data yang tidak relevan. Untuk mengetahui seberapa baik praproses yang digunakan, perlu dilakukan *spell checking* pada setiap *tweet* yang digunakan untuk menghitung berapa banyak kata yang dapat diubah menjadi kalimat baku (Clark *and* Araki, 2011). Proses *spell checking* dilakukan dua kali yaitu dengan menggunakan modifikasi karakter berulang dan tanpa modifikasi karakter berulang untuk digunakan sebagai *variable* dalam *paired t-test*. Kemudian dari hasil *spell checking* dilakukan pengujian *paired t-test* yang ditunjukkan dalam persamaan 3.1. Dalam persama ini, parameter t, D, dan N berturut-turut menyatakan nlai t, selisih nilai sampel tanpa modifikasi dengan nilai sampel dengan modifikasi, dan jumlah sampel.

$$t = \frac{\sum D}{\sqrt{\frac{N\sum D^2 - (\sum D)^2}{N-1}}} \quad (3.1)$$

Contoh penerapan pengukuran kinerja dilakukan setelah melakukan proses *spell checking* yang disajikan pada Tabel 3.10. Setelah itu dilakukan perhitungan nilai D dan D^2 pada masing-masing *tweet* seperti pada Tabel 3.11 yang kemudian dilakukan perhitungan nilai t dengan persamaan 3.1 dengan nilai signifikansi sebesar 5% dan pengujian hipotesis H0 (tidak ada perbedaan kinerja praproses antara tanpa modifikasi dan dengan modifikasi) dan H1 (ada peningkatan kenerja praproses setelah dimodifikasi dibanding tidak dimodifikasi).

Tabel 3.10 Hasil Proses S*pell Checking* Dengan Kata Baku Pada *Tweet*

Tweet	**Tanpa modifikasi**	**Dengan modifikasi**
1	5	6
2	7	7
3	3	3
4	4	6
5	3	3
6	5	6
7	4	4
8	2	4
9	4	5
10	4	4

Tabel 3.11 Hasil Perhitungan D dan D^2

Tweet	**Tanpa modifikasi**	**Dengan modifikasi**	**D**	**D^2**
1	5	6	1	1
2	7	7	0	0
3	3	3	0	0
4	4	6	2	4
5	3	3	0	0
6	5	6	1	1
7	4	4	0	0
8	2	4	2	4
9	4	5	1	1
10	4	4	0	0
Jumlah			7	11

$$t = \frac{7}{\sqrt{\frac{10 \times 11 - 7^2}{10 - 1}}}$$

$$t = -2.688$$

Dari hasil perhitungan t didapatkan nilai t hitung sebesar -2.688 yang akan dibandingkan dengan nilai t tabel 2.262 sehingga |2.688| > |2.262| dapat disimpulkan H0 ditolak dan H1 diterima. Dengan demikian ada peningkatan kenerja praproses setelah dimodifikasi dibanding tidak dimodifikasi.

3.5. Proses Klasifikasi

Setelah dilakukan praproses, setiap dokumen hanya memiliki token-token yang memiliki makna dan relevan dengan penelitian sehingga dokumen siap untuk dilakukan proses klasifikasi. Pada tahap ini, metode yang digunakan adalah SVM.

Dari beberapa algoritma klasifikasi banyak penelitian yang telah membuktikan keunggulan yang dimiliki SVM dengan memiliki nilai akurasi yang tinggi jika dibandingkan dengan algoritma yang lain. Pada penelitian yang dilakukan Vidya, dilakukannya pengujian klasifikasi sentimen dengan menggunakan SVM, NB, dan DT. Hasil penelitiannya menunjukan kinerja SVM lebih unggul dari NB dan DT (Vidya *et al*., 2015). SVM banyak diterapkan dalam konteks klasifikasi terutama klasifikasi dengan sumber data berupa teks dan telah dibuktikan oleh banyak penelitian (Arifiyanti, 2015). Meskipun data yang didapatkan dari sosial media seperti twitter memiliki karakteristik yang unik, SVM dapat mencapai akurasi yang tinggi untuk mngklasifikasikan sentimen saat menggabungkan fitur yang berbeda (Akaichi, 2013). Begitu pula dengan penelitian yang dilakukan (Amolik *et al*., 2016), ia melakukan klasifikasi sentimen dengan membandingkan SVM dan NB dengan nilai akurasi SVM lebih unggul dari NB. Dalam penelitian ini proses kalasifikasi menggunakan algoritma SVM dalam melakukan proses klasifikasi.

Pada penelitian ini kernel yang digunakan adalah kernel linear. Penggunaan kernel linear disebabkan pada penelitian yang dilakukan oleh *(Hsu et al*., 2016), ia berpendapat bahwa penggunaan kernel linier pada SVM lebih cepat dari pada

menggunakan kernel lainnya dalam klasifikasi teks, baik jumlah dokumen dan kata yang besar. Sedangkan menurut (Joachims, 2005) berpendapat bahwa sebagian besar permasalahan klasifikasi teks dipisahkan secara linear. Penerapan metode ini dilakukan dengan menggunakan tools WEKA yang merupakan peangkat lunak yang menyediakan berbagai macam algoritma untuk penggalian data dengan konfigurasi seperti yang dijelaskan pada Tabel 3.12. Sebelum melakukan klasifikasi dengan menggunakan tools WEKA, dokumen dirubah menjadi bentuk *.arff* terlebih dahulu.

Tabel 3.12 Konfigurasi SVM pada WEKA

Konfigurasi	
Kernel type	linear
gamma	0
coef	0
cost	1
degree	3
eps	0.001
gamma	0
seed	1
loss	0.1
nu	0.5

3.6. Uji Coba dan Analisis Hasil

Pada bagian ini akan dijelaskan mengenai skenario uji coba dan analisisnya. Skenario uji coba ini merupakan rencana uji coba sehingga analisis dari uji coba yang dilakukan dapat menjawab rumusan masalah dan tujuan yang telah ditetapkan sebelumnya.

3.6.1 Skenario Uji Coba

Penelitian ini terfokus pada tahap penghapusan karakter pada tahap praproses. Oleh karena itu dalam penelitian ini akan dilakukan beberapa tahap pengujian kemudian hasil klasifikasinya saling dibandingkan.

Pengukuran kinerja menggunakan metode *k-fold cross validation* dimana data dibagi sejumlah 10 *folds* (Basari *et al*., 2013), kemudian proses *testing* dan *training* dilakukan sebanyak 10 kali dengan membagi keseluruhan data menjadi 10

kelompok yang secara bergantian akan dilakukan pengujian dengan 9 data latih dan 1 data data uji hingga hasil akurasi keluar. Proses pengukuran kinerja menggunakan pengukuran akurasi, presision, *recall*, dan *f-measure* untuk mengetahui seberapa akurat model klasifikasi dalam melakukan klasifikasi sentimen *tweet*s. Hasil akurasi tersebut didapat dari rata-rata akurasi pada setiap iterasi.

Tahap pengujian dalam penelitian ini dibagi menjadi dua tahap. Pengujian pertama dilakukan untuk mengetahui peran dari tahapan penghapusan karakter berulang. Pengujian ini dilakukan dengan cara melakukan perbandingan dengan menggunakan tahapan penghapusan karakter berulang dan tidak menggunakan tahapan penghapusan karakter berulang pada. Dengan melaukan pengujian ini diharapkan dapat mengetahui peran dari tahap penghapusan karakter berulang terhadap hasil klasifikasi.

Pengujian kedua dilakukan untuk mengetahui kinerja dari tahapan penghapusan karakter berulang yang diusulkan. Pegujian ini dilakukan dengan membandingkan tahapan penghapusan karakter berulang dengan menggunakan modifikasi dan tidak menggunakan modifikasi. Pengujian ini dilakukan untuk mengetahui kinerja klasifikasi dengan dilakukannya modifikasi tahap penghapusan karakter berulang.

3.6.2 Analisis Hasil Uji Coba

Berdasarkan hasil percobaan yang telah dirancang pada skenario uji coba, kemudian akan dilakukan analisis untuk mengetahui performa dari pengklasifikasi, performa dari tahap penghapusan karakter berulang yang diusulkan, dan performa dari algoritma yang digunakan untuk pembobotan kata. Dari hasil pengujian yang telah dilakukan akan saling dibandingkan untuk mengetahui kinerja tahap penghapusan karakter yang diusulkan.

Halaman ini sengaja dikosongkan

BAB IV
UJI COBA DAN ANALISIS HASIL

Dalam bab ini akan dijelaskan mengenai proses uji coba pada rancangan yang diusulkan, kemudian akan dilakukan analisis untuk mengetahui kinerja dari rancangan yang diusulkan.

4.1. Penyiapan Data

Tahap penyiapan data merupakan tahapan untuk mempersiapkan data yang digunakan dalam penelitian ini. Proses yang dilakukan pada tahapan ini meliputi proses pengumpulan data, dan pengelompokan sentiment.

4.1.1. Pengumpulan Data

Pengambilan data *tweet* yang digunakan dalam penelitian ini dimulai pada tanggal 23 Januari 2017 hingga 17 Februari 2017. Data *tweet* yang mengandung sentimen terkumpul sebanyak 5840 *tweet*. Dari 5840 tweet yang telah terkumpul terdapat 8022 kata memiliki perulangan karakter.

4.1.2. Pengelompokan Sentimen

Setelah data diperoleh kemudian dilakukan pengelompokan *tweet* berdasarkan sentimen yang terkandung didalamnya. Pengelompokan dilakukan dengan membagi tiga kelas sentimen yaitu kelas sentimen positif, negatif, dan netral. Pengelompokan ini dilakukan secara manual berdasarkan jenis opininya. Proses pengelompokan sentimen menghasilkan jumlah *tweet* positif, negatif, dan netral sebayak 917, 4923, dan 879 *tweet*.

Pada data twitter yang diperoleh dalam penelitian ini terjadi ketidak seimbangan pada masing-masing kelas sentimen. Dari hasil pengelompokan sentiment dapat dilihat data yang mengandung sentimen negatif lebih banyak daripada data yang mengandung sentimen positif atau netral. Hal itu dikarenakan data yang diambil dalam penelitian ini adalah tweet dari akun layanan pengaduan atau *customer service* pada penyedia layanan telekomunikasi seluler di Indonesia. Oleh karena itu perlu dilakukan teknik *stratified sampling* seperti pada penelitian

yang dilakukan oleh (Vidya et al., 2015) untuk mendapatkan data yang seimbang sebanyak 2400 *tweet* (kelas positif, 800 kelas negatif, dan 800 kelas netral). Dari 2400 *tweet* terdapat 1163 kata memiliki perulangan karakter dengan 57% diantaranya adalah perulangan karakter yang tidak dapat dikenali kata baku dan 43% adalah perulangan karakter yang dikenali kata baku.

4.2. Lingkungan Uji Coba

Lingkungan uji coba merupakan kriteria perangkat pengujian yang digunakan dalam menguji sistem yang dibangun pada tugas akhir ini. Lingkungan uji coba terdiri dari perangkat keras dan perangkat lunak. Adapun perangkat keras yang digunakan ditunjukkan pada Tabel 4.1. Selain perangkat keras juga digunakan beberapa aplikasi perangkat lunak untuk uji coba dalam penelitian ini yang ditunjukkan pada Tabel 4.2.

Tabel 4.1 Spesifikasi Perangkat Keras Lingkungan Uji Coba

Perangkat Keras	Spesifikasi
Jenis	Laptop
Processor	Intel® Core™ i7 4720HQ Processor
Memory (RAM)	DDR3L 1600 MHz 8 GB

Tabel 4.2 Spesifikasi Perangkat Lunak Lingkungan Uji Coba

Perangkat Lunak	Spesifikasi
Sistem Operasi	Windows 10
Bahasa Pemrograman	PHP
Framework	Codeigniter
Tools	Xampp v3.2.1 PhpStrom v2017.1.2 Weka v3.6 R v3.3.2

4.3. Praproses Teks

Setelah data *tweet* terkumpul kemudian dilakukan tahap praproses. Tahap ini dilakukan untuk menghilangkan derau agar data yang akan diolah sesuai dengan kebutuhan penelitian. Selain itu praproses dilakukan untuk mempersiapkan data mentah agar dapat diolah pada proses selanjutnya. Praproses yang dilakukan dalam penelitian ini dilakukan menggunakan aplikasi yang telah dibangun. Langkah pertama adalah *tweet* dipecah menjadi beberapa token dengan menggunakan

whitespace (“ “) sebagai pemisahnya. Kemudian dilakukan data yang tidak relevan dengan penelitian seperti simbol hashtag, retwett, URL, dan akun akan dihapus. Dapat dilihat pada Tabel 4.3 bahwa proses pembersihan derau membersihkan nama akun twitter “@XLCare” dan “@myXL”. Langkah ketiga adalah kata akan dirubah menjadi huruf kecil dan karakter selain huruf akan dihapus. Dapat dilihat bahwa angka '2' dan karakter 'M' hilang setelah melalui tahap *case folding.* Setelah itu dilakukan penghapusan karakter berulang jika ditemukan terdapat kata yang mengalami perulangan didalamnya. Dapat dilihat bahwa kata “gangguan” dapat diperbaiki menjadi “ganggu”. Langkah kelima kata yang terdaftar dalam *stop list* akan dihapus. Kemudian kata akan dirubah menjadi kata bakunya dan tahap terahir kata yang tidak sesuai dengan kata baku akan dirubah menjadi kata baku seperti “lg”, “gmn”, “udh”, dan “gk” menjadi “lagi”, “gimana”, “sudah”, dan “tidak”.

Tabel 4.3 Contoh Tahapan Praproses Teks

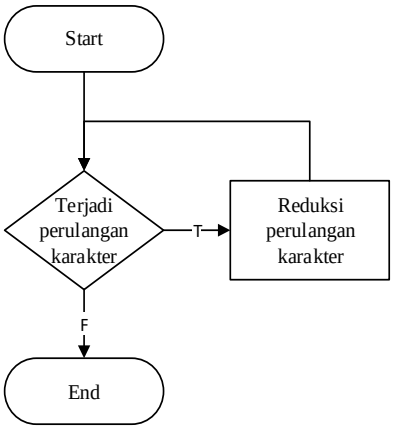
Tahap	Kata
Masukan	Min @XLCare lg gangguan atw gmn @myXL udh 2jm gk da sinyal smskl
Tokenizing	“Min“ ,”@XLCare”, “lg”, “gangguan”, “atw”, “gmn”, “@myXL”, “udh”, “2jm”, “gk”, “da”, “sinyal”, “smskl”
Normalisasi	“Min”, “lg”, “gangguan”, “atw”, “gmn”, “udh”, “2jm”, “gk”, “da”, “sinyal”, “smskl”
Casefolding	“min”, “lg”, “gangguan”, “atw”, “gmn”, “udh”, “jm”, “gk”, “da”, “sinyal”, “smskl”
Penghapusan karakter berulang	“min”, “lg”, “ganggu”, “atw”, “gmn”, “udh”, “jm”, “gk”, “da”, “sinyal”, “smskl”
Penghapusan kata henti	“min”, “lg”, “ganggu”, “atw”, “gmn”, “udh”, “jm”, “gk”, “da”, “sinyal”, “smskl”
Stemming	“min”, “lg”, “ganggu”, “atw”, “gmn”, “udh”, “jm”, “gk”, “da”, “sinyal”, “smskl”
Konversi kata tidak baku	“min”, “lagi”, “ganggu”, “atw”, “gimana”, “sudah”, “jm”, “tidak”, “da”, “sinyal”, “smskl”

Tabel 4.3 Contoh Tahapan Praproses Teks (lanjutan)

Tahap	Kata
Keluaran	min lagi ganggu atw gimana sudah jm tidak da sinyal smskl

4.3.1. Penghapusan Karakter Berulang

Pada penelitian yang dilakukan (Illecker, 2015), proses penghapusan karakter berulang dilakukan dengan cara mencari karakter berulang yang ditemukan pada setiap karakter, kemudian dilakukan penghapusan karakter tersebut sehingga tidak terjadi perulangan seperti yang dijelaskan pada Gambar 4.1. Namun terjadi kesalahan ketika memproses kata yang sudah dalam bentuk kata baku seperti "hingga" menjadi "hinga", "saat" menjadi "sat" (Illecker, 2015). Kata akan kehilangan makna sehingga tidak dapat diproses pada tahap selanjutnya. Selain itu penggunaan data twitter setiap kata yang memiliki makna sangat berharga karena terdapat batasan jumlah karakter tiap *tweet*. Sehingga jika ada kelasahan pemprosesan yang mengakibatkan hilangnnya makna dalam sebuah kata maka akan berdampak besar pada hasil dari klasifikasi. Hal ini perlu dilakukan karena terdapat keterbatasan jumlah karakter pada data twitter (140 karakter), sehingga sangat disayangkan jika terdapat karakter yang memiliki makna tidak dapat dikenali karena terdapat perulangan karakter.



Gambar 4.1 Alur Proses Penghapusan Karakter Berulang

4.3.2. Jenis Perulangan Karakter

Jika dikemompokkan berdasarkan jenis perulangan yang terjadi pada setiap kata dapat dibedakan menjadi empat macam yaitu kata baku mengandung perulangan yang mengalami perulangan karakter lebih dari satu jenis karakter, kata baku mengandung perulangan yang tidak mengalami perulangan karakter, kata baku tidak mengandung perulangan yang mengalami perulangan karakter, dan kata baku tidak mengandung perulangan yang mengalami perulangan karakter lebih dari satu jenis karakter. Setiap perulangan tersebut memiliki ciri-ciri yang berbeda sehingga memerlukan perlakuan yang berbeda untuk mengekstrak menjadi kata yang dapat dikenali. Empat jenis perulangan tersebut disajikan pada Tabel 4.4.

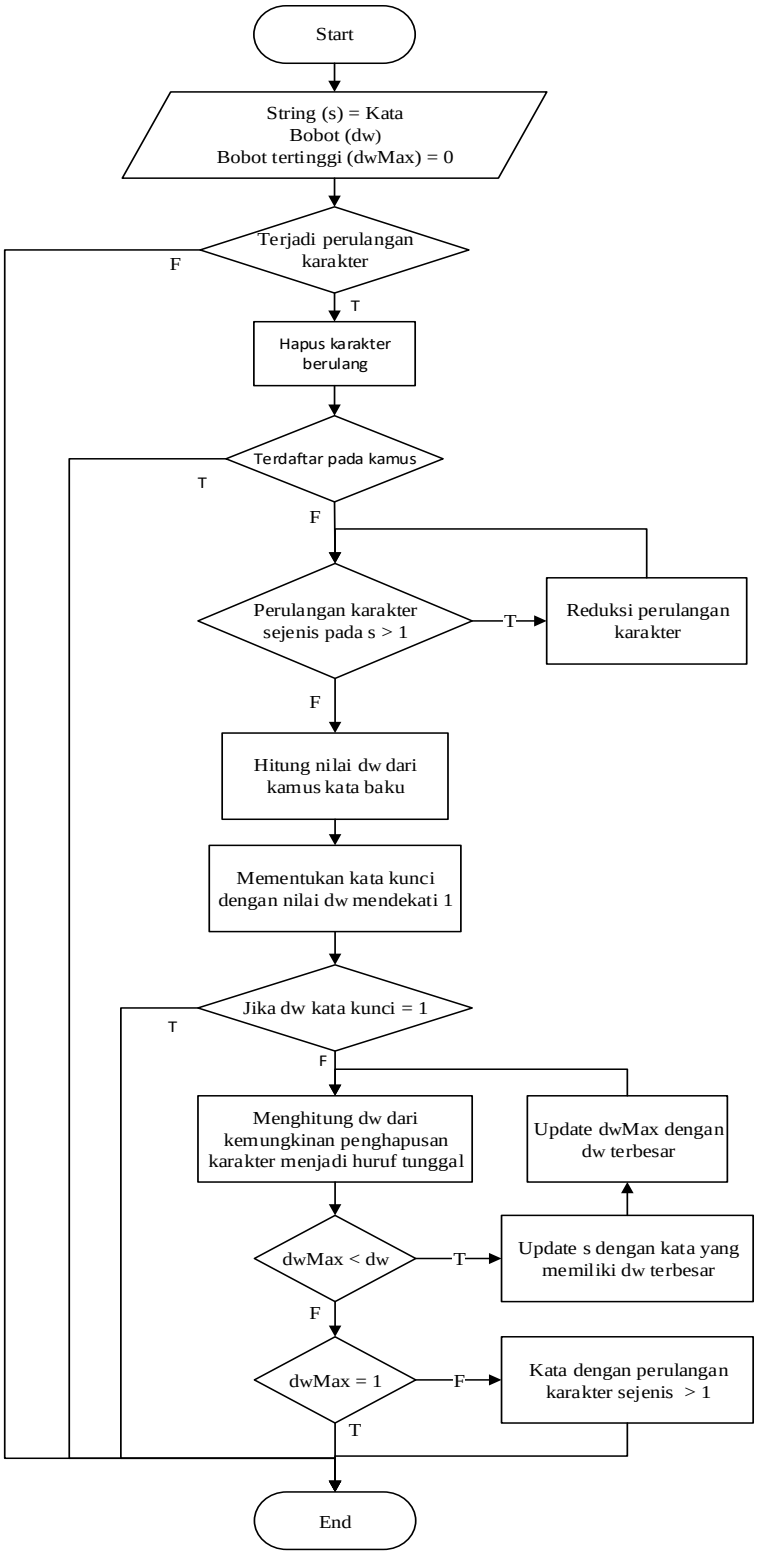
Tabel 4.4 Jenis Perulangan Karakter

No	Jenis Perulangan	Contoh Kata
1	Kata baku mengandung perulangan yang mengalami perulangan karakter lebih dari satu jenis karakter	“pelanggannya”, “mengganggu”, “penggunaan”, “pembukaannya”, “berlangganaan”
2	Kata baku mengandung perulangan yang tidak mengalami perulangan karakter	“maaf”, “manfaat”, “berlangganan”, “hingga”, “panggil”, “saat”
3	Kata baku tidak mengandung perulangan yang mengalami perulangan karakter	“kecewaaa”, “lagiii”, “payaaaaaaaaaah”, “terusss”, “jauhhhh”, “cobaan”
4	Kata baku tidak mengandung perulangan yang mengalami perulangan karakter lebih dari satu jenis karakter	“buseeettttttt”, “hhilaaanngg”, “masiiihhh”, “kenyataannya”, “pertanyaannya”

4.3.3. Modifikasi Penghapusan Karakter Berulang

Dalam penelitian ini, dilakukan pengembangan proses penghapusan karakter berulang dari Gambar 4.1 yang dilakukan oleh (Illecker, 2015). Pengembangan proses tersebut membutuhkan kamus kata baku Bahasa Indonesia dan penggunaan algoritma Jaro Winkler pada persamaan (2.2) yang digunakan utuk mengukur tingkat kesamaan (*similarity*) antar dua kata (*string).* Kamus kata baku digunakan

sebagai pengetahuan (*knowledge*) sistem untuk mengetahui macam kata baku Bahasa Indonesia. Alur pengembangan proses penghapusan karakter berulang dijelaskan pada Gambar 4.2.



Gambar 4.2 Pengembangan Proses Penghapusan Karakter Berulang

Proses penghapusan karakter berulang dimulai dengan memeriksa apakah didalam kata mengalami perulangan. Setiap kata akan diperiksa terlebih dahulu apakah kata tersebut mengandung perulangan karakter yang sejenis (“sehingga”, ”terusss”, “berlanggnaan”, “masiiihhh”) atau tidak. Jika ditemukan perulangan dilanjutkan ke tahap ke dua yaitu dilakukan pencarian kesamaan kata dengan kamus Bahasa Indonesia untuk menentukan metode penghapusan yang akan dilakukan. Jika pada tahap ini kata berhasil dikenali maka perulangan akan dihapus pada kata, namun jika kata tidak dikenali oleh kamus maka akan dilanjutkan ke tahap berikutnya yaitu tahap reduksi perulangan. Kata yang melewati tahap ini adalah kata yang memiliki perulangan karakter sejenis lebih dari satu perulangan seperti “sehingga” menjadi “sehingga”, ”terusss” menjadi ”teruss”, “berlanggnaan” menjadi “berlanggnaan”, “masiiihhh” menjadi “masiihh”. Pada tahap empat, hasil dari tahap tiga digunakan untuk mencari kunci pada kamus kata baku dengan menghitung tingkat kesamaan (*similarity*). Pencarian kata kunci ini digunakan untuk mencari kemiripan kata dari keluaran tahap tiga dengan kata baku yang ada didalam kamus kata baku. Proses ini menggunakan algoritma *Jaro Winkler* untuk mendapatkan nilai bobot tiap kata baku yang ada didalam kamus. Tahap berikutnya dilakukan setelah setiap kata pada kamus memiliki bobot, kemudian dipilih kata yang memiliki bobot mendekati nilai 1 yang digunakan sebagai kunci pada tahap selanjutnya. Tahap enam adalah menghitung bobot dari kemungkinan penghapusan karakter menjadi huruf tunggalnya. Seperti kata “masiihh” penghapusan karakter yang mungkin dilakukan adalah “masiih” dan “masihh” yang masing-masing dihitung kesamaannya dengan kata kunci yang telah ditetapkan pada tahap lima. Dari proses perhitungan kesamaan yang menghasilkan nilai bobot masing-masing dari kemungkinan penghapusan karakter, kemudian dipilih bobot yang mendekati nilai 1 yang digunakan untuk masukan pada tahap lima. Perulangan ini akan selesai hingga tidak ditemukannya bobot yang lebih besar dari bobot maksimum sebelumnya. Setelah perulangan selesai, jika bobot maksimum sama dengan 1 menandakan bahwa ditemukannya kata baku yang kemiripannya 100% dengan masukan kata pada proses ini sehingga keluaran pada proses ini adalah kata dengan bobot 1. Namun, jika bobot maksimum lebih kecil dari 1 menandakan bahwa tidak ditemukan kemiripan dengan kamus Bahasa Indonesia, sehingga keluaran dari

proses ini adalah kata dengan bobot maksimum untuk mengurangi keanekaragaman pada kata yang sama.

Inti dari modifikasi ini adalah jika ditemukan perulangan karakter pada kata, maka penghapusan karakter tidak langsung dilakukan dengan mereduksi karakter berulang menjadi karakter tunggal yang menyebabkan kata kehilangan makna dikarenakan mereduksi karakter yang berlebihan. Oleh karena itu pada modifikasi yang dilakukan, karakter akan direduksi hingga menyisahkan satu perulangan. Hal ini dilakukan untuk menghindari reduksi perulangan yang berlebihan yang akan membuat beberapa kata baku yang mengandung perulangan kehilangan makna seperti persoalan pertama dan kedua pada Tabel 4.4. Jika tahap ini tidak dilakukan maka kata seperti "sehingga", perulangan karakter "g" akan direduksi menjadi karakter tunggal yang menyebabkan kata "sehinga" tidak dapat diproses dengan baik karena kehilangan makna.

Contoh penerapan pengembangan proses penghapusan karakter berulang dengan masukkan pada macam-macam jenis perulangan pada Tabel 4.4 dengan menggunakan kamus kata baku seperti pada Tabel 4.5:

Tabel 4.5 Contoh Kamus Kata Baku

No	Kata Baku
1	Terus
2	Masih
3	Sayat
4	Hingga
5	Hinggap
6	Sehingga
7	Bangga
8	Tangga
9	Enggan
10	Anyam
11	Langgan

1. Masukan = "berlangganaan"
    a. Tahap 1, pada tahap ini dilakukan pemeriksaan apakah didalam kata mengalami perulangan atau tidak. Kata "berlangganaan" mengalami perulangan pada karakter "g" dan "a" sehingga memenuhi syarat dan dilanjutkan pada tahap dua.

b. Tahap 2, pada tahap ini perulangan pada kata akan dihapus menjadi “berlanganan” lalu dicocokan dengan kamus Bahasa Indonesia. Karena kata “berlanganan” tidak ditemukan pada kamus sehingga dilanjutkan ke tahap tiga.
c. Tahap 3, pada tahap ini dilakukan pemeriksaan jumlah perulangan yang terkandung pada kata tersebut. Kata “berlangganaan” mengalami perulangan pada karakter “g” sebanyak satu perulangan dan karakter “a” mengalami satu perulangan. Karena tidak ditemukan perulangan karakter sejenis > 1, kemudian dilanjutkan pada tahap empat.
d. Tahap 4, pada tahap ini dilakukan pencarian kesamaan kata “berlangganaan” dengan kamus Bahasa Indonesia. Setiap kata baku pada kamus dibandingakan dengan kata “berlangganaan” dan dicari tingkat kemiripannya dengan menggunakan persamaan (2.2). Hasil perhitungan kemiripan disajikan pada Tabel 4.6.

Tabel 4.6 Perhitungan Kemiripan Kata “berlangganaan”

No	Kata Baku	Bobot
1	terus	0.517
2	masih	0
3	sayat	0.425
4	hingga	0.533
5	hinggap	0.501
6	sehingga	0.569
7	bangga	0.838
8	tangga	0.739
9	enggan	0.533
10	anyam	0.425
11	langgan	0.846

e. Tahap 5, setelah setiap kata baku pada kamus dibandingakan dan memiliki nilai bobot. Kemudian ditentukan nilai bobot yang mendekati dengan 1 sebagai kunci pada tahap berukitnya yaitu kata “langgan” pada kamus kata baku dengan bobot sebesar 0.846.
f. Tahap 6, pada tahap ini melakukan perhitungan bobot dari kemungkinan penghapusan karakter “berlangganaan” menjadi

huruf tunggalnya yaitu “berlangnaan” sebesar 0.785 dan “berlangganan” sebesar 0.861. Karena bobot karakter “berlangganan” sebesar 0.861 lebih besar dari bobot kunci sebesar 0.846, sehingga dilakukan perhitungan kembali pada tahap enam dengan masukan kata “berlangganan”. Setelah dihitung kemungkinan penghapusan karakter “berlangganan” menjadi huruf tunggalnya yaitu “berlanganan” sebesar 0.800. Tahapan ini tidak mengalami perulangan kembali karena tidak ditemukannya kemungkinan penghapusan karakter dan bobot dari penghapusan kemungkinan karakter tidak ada yang melebihi bobot kunci, sehingga tahapan ini tidak mengalami perulangan kembali dan dilanjutkan pada tahap berikutnya.

g. Tahap 7, karena bobot maksimal yang didapatkan < 1, sehingga keluaran dari tahapan penghapusan berulang adalah keluaran dari tahap 3 yaitu kata “berlangganan”.

2. Masukan = “sehingga”
    a. Tahap 1, pada tahap ini dilakukan pemeriksaan apakah didalam kata mengalami perulangan atau tidak. Kata “sehingga” mengalami perulangan pada karakter “g” sehingga memenuhi syarat dan dilanjutkan pada tahap dua.
    b. Tahap 2, pada tahap ini perulangan pada kata akan dihapus menjadi “sehinga” lalu dicocokan dengan kamus Bahasa Indonesia. Karena kata “sehinga” tidak ditemukan pada kamus sehingga dilanjutkan ke tahap tiga.
    c. Tahap 3, pada tahap ini dilakukan pemeriksaan jumlah perulangan yang terkandung pada kata tersebut. Kata “sehingga” mengalami perulangan pada karakter “g” sebanyak satu perulangan. Karena tidak ditemukan perulangan karakter sejenis > 1, kemudian dilanjutkan pada tahap empat.
    d. Tahap 4, pada tahap ini dilakukan pencarian kesamaan kata “sehingga” dengan kamus Bahasa Indonesia. Setiap kata baku pada kamus dibandingakan dengan kata “sehingga” dan dicari

tingkat kemiripannya dengan menggunakan algoritma persamaan (2.2). Hasil perhitungan kemiripan disajikan pada Tabel 4.7.

Tabel 4.7 Perhitungan Kemiripan Kata "sehingga"

No	Kata Baku	Bobot
1	terus	0.441
2	masih	0.547
3	sayat	0.497
4	hingga	0.916
5	hinggap	0.869
6	sehingga	1
7	bangga	0.722
8	tangga	0.722
9	enggan	0.829
10	anyam	0
11	langgan	0.690

e. Tahap 5, setelah setiap kata baku pada kamus dibandingakan dan memiliki nilai bobot. Kemudian ditentukan nilai bobot yang mendekati dengan 1 sebagai kunci pada tahap berukitnya yaitu kata "sehingga" pada kamus kata baku dengan bobot sebesar 1.
f. Tahap 6, pada tahap ini melakukan perhitungan bobot dari kemungkinan penghapusan karakter "sehingga" menjadi huruf tunggalnya yaitu "sehinga" sebesar 0.975. Tahapan ini tidak terjadi perulangan karena bobot dari kemungkinan penghapusan karakter berulang tidak lebih besar dari bobot kunci sebesar 1 sehingga dilanjutkan pada tahap berikutnya.
g. Tahap 7, bobot maksimal yang didapatkan adalah 1 sehingga keluaran dari tahapan penghapusan berulang adalah kata "sehingga".

3. Masukan = "terusss"
    a. Tahap 1, pada tahap ini dilakukan pemeriksaan apakah didalam kata mengalami perulangan atau tidak. Kata "terusss" mengalami perulangan pada karakter "s" sehingga memenuhi syarat dan dilanjutkan pada tahap dua.

b. Tahap 2, pada tahap ini perulangan pada kata akan dihapus menjadi “terus” lalu dicocokan dengan kamus Bahasa Indonesia. Karena kata “terus” ditemukan pada kamus sehingga keluaran dari tahapan penghapusan berulang adalah kata “terus”.

4. Masukan = “masiiihhh”
    a. Tahap 1, pada tahap ini dilakukan pemeriksaan apakah didalam kata mengalami perulangan atau tidak. Kata “masiiihhh” mengalami perulangan pada karakter “i” dan “h” sehingga memenuhi syarat dan dilanjutkan pada tahap dua.
    b. Tahap 2, pada tahap ini perulangan pada kata akan dihapus menjadi “masih” lalu dicocokan dengan kamus Bahasa Indonesia. Karena kata “terus” ditemukan pada kamus sehingga keluaran dari tahapan penghapusan berulang adalah kata “masih”.

Dari uji coba diatas, pengembangan penghapusan karakter berulang berhasil menyelesaikan dari empat jenis perulangan pada kata tanpa menghilangkan makna dari kata tersebut. Berbeda dengan penghapusan karakter sebelumnya yang dapat menghilangkan makna dari karakter yang dijelaskan pada Tabel 4.8.

Tabel 4.8 Perbandingan Pengembangan Penghapusan Karakter Berulang

No	Jenis Perulangan	Masukan	Keluaran	
			Metode Illecker	**Modifikasi penghapusan katakter berulang**
1	Kata baku mengandung perulangan yang mengalami perulangan karakter lebih dari satu jenis karakter	berlangga naan	~~berlanganan~~	berlangganan
2	Kata baku mengandung perulangan yang tidak mengalami perulangan karakter	sehingga	~~sehinga~~	sehingga
3	Kata baku tidak mengandung perulangan yang mengalami perulangan karakter	terusss	terus	terus

Tabel 4.8 Perbandingan Pengembangan Penghapusan Karakter Berulang (lanjutan)

No	Jenis Perulangan	Masukan	Keluaran	
			Metode Illecker	Modifikasi penghapusan katakter berulang
4	Kata baku tidak mengandung perulangan yang mengalami perulangan karakter lebih dari satu jenis karakter	masiiihhh	masih	masih

4.4. Skenario Uji Coba

Pada uji coba penelitian ini terdapat tiga skenario uji coba. Uji coba yang dilakukan diantaranya adalah uji coba yaitu uji coba modifikasi penghapusan karakter berulang, ujicoba perbandingan performa klasifiaksi, dan uji coba perbandingan penghapusan kata henti.

4.4.1. Uji Coba Modifikasi Penghapusan Karakter Berulang

Untuk menguji apakah metode dan cara yang diimplementasikan sudah dilakukan dengan benar perlu dilakukan perhitungan secara manual. Uji coba ini dilakukan untuk mengetahui seberapa baik hasil yang didapatkan setelah dilakukannya modifikasi pada penghapusan karakter berulang. Pengujian dilakukan dengan membandingkan jumlah kata yang dapat dikenali dengan baik tanpa menggunakan penghapusan karakter berulang, menggunakan penghapusan karakter berulang, dan menggunakan modifikasi penghapusan karakter berulang.

Uji coba ini dilakukan dengan memberikan masukan berupa sebuah kata yang memiliki kesalahan penulisan perulangan. Pengujian dilakukan tiga kali dengan skenario yang berbeda yaitu pengujian dengan memberikan masukan berupa kata yang tidak memiliki perulangan, kata yang mengandung perulangan, dan kata yang mengalami kesalahan dalam perulangan sehingga tidak dapat dikenali. Skenario pertama dilakukan dengan memberikan masukan berupa kata yang tidak memiliki perulangan karakter. Skenario ini bertujuan untuk menguji apakah modifikasi yang dilakukan dapat mengenali kata dengan baik yang tidak memiliki perulangan

didalamnya dan keluaran yang dihasilkan sesuai dengan tanpa dilakukannya penghapusan karakter berulang. Skenario kedua dilakukan dengan memberikan masukan berupa kata yang mengandung perulangan karakter. Skenario ini ditujukan untuk menguji apakah modifikasi yang dilakukan dapat memproses kata yang mengalami perulangan karakter dan hasil yang diperoleh dapat dikenali dengan baik setelah dilakukan penghapusan karakter yang berlebih. Skenario ketiga dilakukan dengan memberikan masukan berupa kata yang mengalami kesalahan dalam perulangan sehingga tidak dapat dikenali jika tidak dilaukan proses penghapusan karakter. Skenario ini dilakukan untuk menguji apakah modifikasi yang dilakukan dapat menangani kata yang mengalami perulangan berlebih menjadi kata yang dapat dikenali dengan baik. Dari hasil penghapusan karakter berulang, kemudian dilakukan proses pencocokan dengan kamus Bahasa Indonesia yang bertujuan untuk menguji hasil penghapusan karakter berulang dapat berjalan dengan baik atau tidak.

Selain itu juga dilakukan perbandingan jumlah kata yang dapat dikenali dan tidak dapat dikenali oleh kamus. Uji coba ini dilakukan untuk mengetahui kinerja dari modifikasi penghapusan karakter berulang pada hasil klasifikasi dibandingkan dengan tanpa dilakukan penghapusan karakter berulang maupun tanpa dilakukan modifikasi penghapusan karakter berulang.

4.4.2. Uji Coba Perbandingan Penghapusan Kata Henti

Pengujian perbandingan penghapusan kata henti dilakukan dengan membandingkan akurasi dari klasifikasi dengan menggunakan dan tidak menggunakan penghapusan kata henti pada tahap praproses. Pengujian ini dilakukan karena terdapat kata yang mengandung sentimen terhapus karena terdaftar pada *stop list*.

4.4.3. Uji Coba Tahapan Praproses

Pengujian dilakukan dengan dengan membandingkan hasil klasifikasi dengan menggunakan tahap penghapusan karakter berulang, stemming, konversi kata tidak baku, dan kombinasi dari seluruh proses. Pengujian *tokenizing*, pembersihan derau, dan *case folding* merupakan tahap yang paling umum dilakukan dan banyak

dibuktikan pada penelitian-penelitian lainnya pada tahap awal praproses sehingga tidak dilakukan pengujian.

4.5. Hasil dan Analisis Uji Coba

Pada bagian ini akan dijelaskan mengenai hasil dari tahap uji coba yang dilakukan dari skenario uji coba yaitu uji coba modifikasi penghapusan karakter berulang, uji coba perbandingan performa klasifiaksi, dan uji coba perbandingan penghapusan kata henti.

4.5.1. Hasil dan Analisis Uji Coba Modifikasi Penghapusan Karater Berulang

Pengujian skenario pertama dilakukan dengan memberikan masukan berupa kata yang tidak memiliki perulangan. Pengujian ini dilakukan dengan membandingkan hasil dari proses tanpa menggunakan penghapusan karakter berulang, menggunakan penghapusan karakter berulang, dan modifikasi penghapusan karakter berulang apakah dapat dikenali dengan baik atau tidak dengan menggunakan 40 kata yang tidak mengalami perulangan. Dari hasil yang didapatkan, masing-masing proses dapat mengatasi dengan baik kata yang tidak memiliki perulangan dengan jumlah kata yang dapat dikenali sebanyak 40 kata sehingga keseluruhan kata dapat dikenali dengan baik seperti pada Tabel 4.9.

Dari skenario pertama dapat dilihat bahwa modifikasi yang dilakukan dapat menyaingi proses lainnya dengan mengatasi kata yang tidak mengalami perulangan. Selain itu juga dapat memproses kata yang mengalami penambahan imbuhan atau akhiran maupun tidak.

Tabel 4.9 Hasil Uji Coba Skenario Pertama

No	Kata Tanpa Perulangan	Penghapusan Karakter Berulang		
		Tanpa Penghapusan	Dengan Penghapusan	Dengan Modifikasi Penghapusan
1	balas	balas	balas	balas
2	jawab	jawab	jawab	jawab
3	cepat	cepat	cepat	cepat
4	lain	lain	lain	lain

Tabel 4.9 Hasil Uji Coba Skenario Pertama (lanjutan)

No	Kata Tanpa Perulangan	Penghapusan Karakter Berulang		
		Tanpa Penghapusan	Dengan Penghapusan	Dengan Modifikasi Penghapusan
5	pelayanan	pelayanan	pelayanan	pelayanan
6	penyelesaian	penyelesaian	penyelesaian	penyelesaian
7	perubahan	perubahan	perubahan	perubahan
8	tidak	tidak	tidak	tidak
9	tanpa	tanpa	tanpa	tanpa
10	bisa	bisa	bisa	bisa
11	berani	berani	berani	berani
12	paksa	paksa	paksa	paksa
13	tolong	tolong	tolong	tolong
14	perlu	perlu	perlu	perlu
15	bulan	bulan	bulan	bulan
16	masih	masih	masih	masih
17	jalan	jalan	jalan	jalan
18	tingkat	tingkat	tingkat	tingkat
29	monoton	monoton	monoton	monoton
30	racun	racun	racun	racun
31	gali	gali	gali	gali
32	gapai	gapai	gapai	gapai
33	gila	gila	gila	gila
34	gapai	gapai	gapai	gapai
35	goyang	goyang	goyang	goyang
36	gagal	gagal	gagal	gagal
37	tulis	tulis	tulis	tulis
38	catat	catat	catat	catat
39	cepat	cepat	cepat	cepat
40	henti	henti	henti	henti
Jumlah		**40**	**40**	**40**

Pengujian skenario kedua dilakukan dengan memberikan masukan berupa kata yang mengandung perulangan seperti pada jenis perulangan satu dan dua pada

Tabel 4.9. Pengujian ini dilakukan dengan membandingkan hasil dari proses tanpa menggunakan penghapusan karakter berulang, menggunakan penghapusan karakter berulang, dan modifikasi penghapusan karakter berulang apakah dapat dikenali dengan baik atau tidak dengan menggunakan 40 kata yang mengandung perulangan. Dari hasil yang didapatkan, dapat dilihat bahwa proses tanpa penghapusan karakter berulang dapat menangani dengan baik dengan jumlah kata yang dapat dikenali sebanyak 40 kata. Namun hasil yang didapatkan pada proses yang menggunakan penghapusan karakter berulang tidak memuaskan. Proses dengan penghapusan karakter berulang tidak mampu untuk menangani kata yang mengandung perulangan didalamnya. Hal itu dikarenakan setiap kata yang mengalami perulangan akan dihapus tanapa memperhatikan makna yang terkandung dalam kata seperti ”anggap” menjadi “angap”, “gangguan” menjadi “ganguan”, dan “perubahannya” menjadi “perubahanya” yang menyebabkan kata akan kehilangan makna dan tidak dapat diproses dengan baik membuat kata tida dapat dikenali. Berbeda dengan proses yang menggunakan modifikasi penghapusan karakter berulang, hasil yang didapatkan memuaskan dengan jumlah kata yang dapat dikenali sebanyak 40 kata. Modifikasi yang dilakukan dapat membedakan kata yang memiliki perulangan dengan kata yang tidak memilii perulangan sehingga membuat modifikasi penghapusan karakter berulang lebih unggul dari penghapusan karakter berulang yang dapat membedakan perlakuan pada kata yang akan diproses menjadi kata dapat dikenali dengan baik tanpa membuat kata kehilangan makna seperti yang dijelaskan pada Tabel 4.10.

Dari skenario kedua dapat dilihat bahwa modifikasi penghapusan karakter berulang yang dilakukan dapat mengungguli proses penghapusan karakter berulang dengan mengatasi kata yang mengalami perulangan. Selain itu juga dapat memproses kata yang mengalami penambahan imbuhan atau akhiran maupun tidak.

Tabel 4.10 Hasil Uji Coba Skenario Kedua

No	Kata Mengandung Perulangan	Penghapusan Karakter Berulang		
		Tanpa Penghapusan	Dengan Penghapusan	Dengan Modifikasi Penghapusan
1	anggap	anggap	~~angap~~	anggap

Tabel 4.10 Hasil Uji Coba Skenario Kedua (lanjutan)

No	Kata Mengandung Perulangan	Penghapusan Karakter Berulang		
		Tanpa Penghapusan	Dengan Penghapusan	Dengan Modifikasi Penghapusan
2	gangguan	gangguan	~~ganguan~~	ganggu
3	sanggup	sanggup	~~sangup~~	sanggup
4	sehingga	sehingga	~~sehinga~~	sehingga
5	sungguh	sungguh	~~sunguh~~	sungguh
6	tanggung	tanggung	~~tangung~~	tanggung
7	tenggelam	tenggelam	~~tengelam~~	tenggelam
8	tetangga	tetangga	~~tetanga~~	tetangga
9	tinggi	tinggi	~~tingi~~	tinggi
10	tunggu	tunggu	~~tungu~~	tunggu
11	unggah	unggah	~~ungah~~	unggah
12	unggul	unggul	~~ungul~~	unggul
13	minggu	minggu	~~mingu~~	minggu
14	tanggal	tanggal	~~tangal~~	tanggal
15	saat	saat	~~sat~~	saat
16	lainnya	lainnya	~~lainya~~	lain
17	perubahannya	perubahannya	~~perubahanya~~	ubah
18	tanggap	tanggap	~~tangap~~	tanggap
19	tinggal	tinggal	~~tingal~~	tinggal
20	tenggara	tenggara	~~tengara~~	tenggara
21	menggali	menggali	~~mengali~~	gali
22	tanggul	tanggul	~~tangul~~	tanggul
23	tangguh	tangguh	~~tanguh~~	tangguh
24	tangga	tangga	~~tanga~~	tangga
25	serangga	serangga	~~seranga~~	serangga
26	punggung	punggung	~~pungung~~	punggung
27	pinggir	pinggir	~~pingir~~	pinggir

Tabel 4.10 Hasil Uji Coba Skenario Kedua (lanjutan)

No	Kata Mengandung Perulangan	Penghapusan Karakter Berulang		
		Tanpa Penghapusan	Dengan Penghapusan	Dengan Modifikasi Penghapusan
28	punggung	punggung	~~pungung~~	punggung
29	penggal	penggal	~~pengal~~	penggal
30	panggung	panggung	~~pangung~~	panggung

Pengujian skenario ketiga dilakukan dengan memberikan masukan berupa kata yang mengalami kesalahan dalam perulangan seperti pada jenis perulangan tiga dan empat pada Tabel 4.9. Pengujian ini dilakukan dengan membandingkan hasil dari proses tanpa menggunakan penghapusan karakter berulang, menggunakan penghapusan karakter berulang, dan modifikasi penghapusan karakter berulang apakah dapat dikenali dengan baik atau tidak dengan menggunakan 20 kata yang tidak mengandung perulangan dari skenario pertama dan 20 kata yang mengandung perulangan dari skenario kedua seperti pada Tabel 4.11.

Dari hasil yang didapatkan, dapat dilihat bahwa proses tanpa penghapusan karakter berulang tidak mampu untuk menangani kata yang mengalami kesalahan dalam pengulangan seperti ”toolooooong“, “cepaaatt“, dan “sanggguup“ yang dikarenakan terdapat perulangan yang menyebabkan kata tidak dapat dikenali dengan baik. Terdapat peningkatan ketika mengunakan proses penghapusan karakter berulang, jumlah kata yang dapat dikenali sebanyak 18 kata. Peningkatan tersebut dikarenakan pada dasarnya kata tersebut memang tidak memiliki perulangan sehingga jika perulangan yang terjadi didalam kata tersebut dihapus maka kata dapat dikenali dengan baik. Namun terdapat 22 kata yang tidak dapat dikenali dengan baik, yaitu kata yang pada dasarnya kata tersebut memang mengandung perulangan seperti “lainnnya” menjadi “lainya”, “angggap” menjadi “angap”, dan “amannyaa” menjadi “amanya” yang mengakibatkan jika perulangan didalam kata tersebut dihapus maka kata tidak dapat dikenali dengan baik. Peningkatan signifikan terjadi menggunakan modifikasi penghapusan karakter berulang. Perulangan yang terjadi pada kata dapat diproses dengan baik, baik kata

mengandung perulangan maupun kata yang tidak mengandung perulangan dapat dikenali dengan baik dengan jumlah kata yang dapat dikenali sebanyak 40 kata. Hal itu disebabkan karena sebelum dilakukan penghapusan karakter, kata akan dicek terlebih dahulu apakah kata tersebut pada dasarnya memang memiliki perulangan atau tidak. Jika kata tersebut pada dasarnya tidak memiliki perulangan, maka perulangan akan dihapus hingga tidak terjadi perulangan. Namun jika pada dasarnya kata tersebut memiliki perulangan, maka penghapusan karakter berulang akan menyisakan satu perulangan yang kemudian akan dilanjutkan pada proses pencarian kemiripan dengan menggunakan Algoritma Jaro Winkler. Kata akan dicari kemiripannya dengan mencari kemungkinan penghapusan karakter berulang hingga menemukan bobot sama dengan 1 yang artinya memiliki kemiripan dengan kamus Bahasa Indonesia 100%. Namun jika tidak menemukan kemiripan, maka hasil dari proses penghapusan yang menyisakan satu perulangan dengan tujuan untuk memper kecil keanekaragaman kata.

Dari skenario ketiga dapat dilihat bahwa modifikasi penghapusan karakter berulang yang dilakukan dapat mengungguli proses lainnya dalam mengatasi kata yang mengalami perulangan maupun tidak. Selain itu juga dapat memproses kata yang mengalami penambahan imbuhan atau akhiran maupun tidak.

Tabel 4.11 Hasil Uji Coba Skenario Ketiga

No	Kesalahan Dalam Perulangan Kata	Penghapusan Karakter Berulang		
		Tanpa Penghapusan	Dengan Penghapusan	Dengan Modifikasi Penghapusan
1	balasss	balasss	balas	balas
2	jawaab	jawaab	jawab	jawab
3	cepaaatt	cepaaatt	cepat	cepat
4	lainnnya	lainnya	~~lainya~~	lain
5	pelayann	pelayann	pelayan	layan
6	penyelesaiaan	penyelesaiaan	penyelesaian	selesai
7	perubahannnya	perubahannnya	~~perubahanya~~	ubah
8	tidaak	tidaak	tidak	tidak
9	taanpaa	taanpaa	tanpa	tanpa
10	bisaaa	bisaaa	bisa	bisa

Tabel 4.11 Hasil Uji Coba Skenario Ketiga (lanjutan)

No	Kesalahan Dalam Perulangan Kata	Penghapusan Karakter Berulang		
		Tanpa Penghapusan	Dengan Penghapusan	Dengan Modifikasi Penghapusan
11	beraani	beraani	berani	berani
12	paksaa	paksaa	paksa	paksa
13	toolooooong	toolooooong	tolong	tolong
14	perluuu	perluuu	perlu	perlu
15	bulann	bulann	bulan	bulan
16	masiiih	masiiih	masih	masih
17	jalaaan	jalaaan	jalan	jalan
18	tingkatt	tingkatt	tingkat	tingkat
19	absenn	absenn	absen	absen
20	amaaan	amaaan	aman	aman
21	angggap	angggap	~~angap~~	anggap
22	ganggguan	ganggguan	~~ganguan~~	ganggu
23	sanggguup	sanggguup	~~sangup~~	sanggup
24	sehinggga	sehinggga	~~sehinga~~	sehingga
25	sungguuuh	sungguuuh	~~sunguh~~	sungguh
26	tangggung	tangggung	~~tangung~~	tanggung
27	tenggelamm	tenggelamm	~~tengelam~~	tenggelam
28	tetanggaaa	tetanggaaa	~~tetanga~~	tetangga
29	tinggiii	tinggiii	~~tingi~~	tinggi
30	tungguuu	tungguuu	~~tungu~~	tunggu
31	ungggah	ungggah	~~ungah~~	unggah
32	unggguul	unggguul	~~ungul~~	unggul
33	mingggu	mingggu	~~mingu~~	minggu
34	tanggaal	tanggaal	~~tangal~~	tanggal
35	saaat	saaat	~~sat~~	saat
36	lainnnya	lainnnya	~~lainya~~	lain
37	amannyaa	amannyaa	~~amanya~~	amannya
38	tangggap	tangggap	~~tangap~~	tanggap

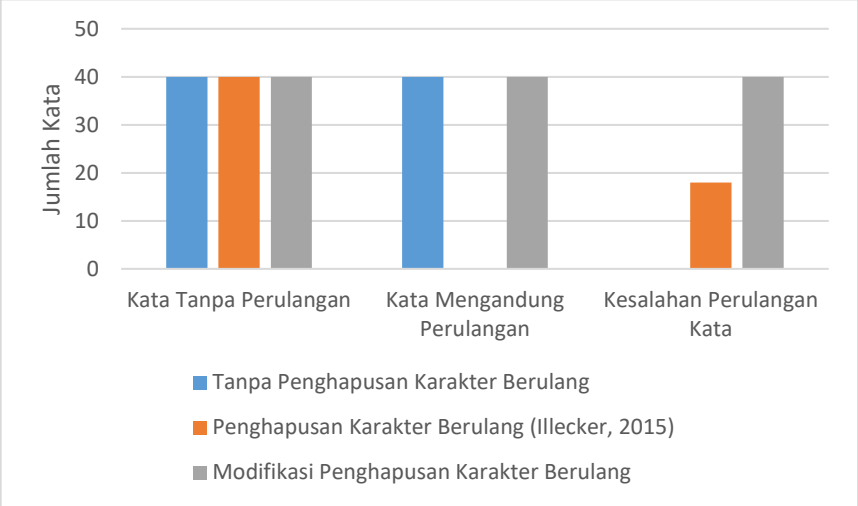
Tabel 4.11 Hasil Uji Coba Skenario Ketiga (lanjutan)

No	Kesalahan Dalam Perulangan Kata	Penghapusan Karakter Berulang		
		Tanpa Penghapusan	Dengan Penghapusan	Dengan Modifikasi Penghapusan
39	tinggaal	tinggaal	~~tingal~~	tinggal
40	tenggaraa	tenggaraa	~~tengara~~	tenggara
Jumlah		**0**	**18**	**40**

Dari hasil pengujian dengan skenario kata tanpa perulangan yang dijelaskan pada Gambar 4.3, keseluruhan kata dapat ditangani dengan baik menggunakan tanpa penghapusan karakter berulang, menggunakan penghapusan karakter berulang maupun modifikasi penghapusan berulang. Sedangkan pada skenario kata mengandung perulangan, kata yang diproses denga tanpa penghapusan karakter berulang dan modifikasi penghapusan karakter berulang dapat dekenali dengan baik, namun pada penghapusan karakter berulang didapatkan hasil yang tidak memuaskan. Hal itu disebabkan karena kata yang harusnya memiliki perulangan akan dihapus menjadi karakter tunggal sehingga tidak dapat dikenali dengan baik. Pada skenario kesalahan perulangan kata, kata yang diproses tanpa menggunakan penghapusan karakter berulang tidak dapat mengenali kata tersebut dengan baik. Hal itu dikarenakan karena terjadi kesalahan penulisan pengulangan kata yang menyebabkan kata tidak dapat dideteksi oleh kamus. Namun ketika dengan menggunakan penghapusan karakter berulang terdapat peningkatan kata yang dapat dikenali sebanyak 18 kata. Hasil terbaik didapatkan dengan menggunakan modifikasi penghapusan karakter berulang dengan jumlah kata yang dapat dikenali dengan baik sebanyak 40 kata.

Hasil dari tiga skenario uji coba penghapusan karakter berulang yang mengahasilkan jumlah kata yang dapat ditangani dengan baik adalah dengan melakukan modifikasi penghapusan karakter berulang yang dirancang dalam penelitian ini. Modifikasi berhasil menangani berbagai jenis perulangan karakter yang ada dalam kata maupun tidak dengan baik. Dari hasil yang dijabarkan pada Gambar 4.3, dapat dilihat bahwa modifikasi penghapusan karakter berulang dapat menangani berbagai jenis perulangan dengan baik pada ketiga skenario diatas.

Pengahapusan karakter berulang biasa baik digunakan ketika terjadi perulangan karakter pada kata yang pada dasarnya tidak memiliki perulangan, sehingga penghapusan karakter menjadi karakter tunggal yang dilakukan membuat menjadikan karakter dapat dikenali dengan baik.



Gambar 4.3 Perbandingan Skenario Penghapusan Karakter Berulang

Berikut merupakan contoh penerapan modifikasi penghapusan karakter berulang dengan contoh *tweet* "Min @XLCare lg gangguan atw gmn @myXL udh 2jm gk da sinyal smskl". **S**etelah melewati tahap *tokenizing*, pembersihan derau, *case folding* seperti pada Tabel 4.3, hasil sementara yang didapatkan adalah "min", "lg", "gangguan", "atw", "gmn", "udh", "jm", "gk", "da", "sinyal", dan "smskl" yang digunakan sebagai Masukan dalam tahap penghapusan karakter berulang. Karena kata "min", "lg", "atw", "gmn", "udh", "jm", "gk", "da", "sinyal", dan "smskl" tidak mengalami perulangan maka tidak dilanjutkan ke tahap berikutnya sehigga keluaran dari masing-masing kata sama seperti masukan pada tahap penghapusan karakter berulang. Namun tidak pada kata "gangguan", karena pada kata "gangguan" mengalami perulangan pada karakter "g" sehingga memenuhi syarat dan dilanjutkan pada tahap dua. Pada tahap dua, perulangan pada kata akan dihapus menjadi "ganguan" lalu dicocokan dengan kamus Bahasa Indonesia. Karena kata "ganguan" tidak ditemukan pada kamus maka dilanjutkan ke tahap tiga. Pada tahap tiga, dilakukan pemeriksaan jumlah perulangan yang terkandung pada kata tersebut. Kata "gangguan" mengalami perulangan pada

karakter “g” sebanyak satu perulangan. Karena tidak ditemukan perulangan karakter sejenis > 1, kemudian dilanjutkan pada tahap empat. Pada tahap empat, sebelum dilakukan pencarian kesamaan kata “gangguan” dengan kamus Bahasa Indonesia, kata akan dicoba untuk dihapus imbuhan “an” yang ada didalam kata menjadi “ganggu” kemudian dilakukan pencocokan dengan kamus Bahasa Indonesia. Untuk mempersingkat pencarian pada kamus, pencarian dilakukan dengan mencari kata dalam kamus yang mengandung awalan “g” dan akhiran “u” dan kemudian cari tingkat kemiripan dengan menggunakan algoritma persamaan (2.1) dan (2.2). Hasil perhitungan kemiripan disajikan pada Tabel 4.12 dengan detail perhitungan pada Tabel 4.13 hingga Tabel 4.17.

Tabel 4.12 Contoh Perhitungan Kemiripan Kata “ganggu” Lima Tertinggi

No	Kata Baku	Bobot
1	ganggu	1
2	gagu	0.911
3	gancu	0.875
4	gandu	0.875
5	gagau	0.857

1. Perhitungan kemiripan kata “ganggu”

Tabel 4.13 Perhitungan Kemiripan “ganggu”

	g	a	n	g	g	u
g	1	0	0	0	0	0
a	0	1	0	0	0	0
n	0	0	1	0	0	0
g	0	0	0	1	0	0
g	0	0	0	0	1	0
u	0	0	0	0	0	1

Dengan:

m = 6

$|s_1|$ = 6

$|s_2|$ = 6

t = 0

l = 6

p = 0.1

Sehingga perhitungan nilai Jaro *distance* dengan persamaan (2.1) adalah:

$$d_j = \frac{1}{3} \times \left(\frac{6}{6} + \frac{6}{6} + \frac{6-0}{6}\right) = 1$$

Dan nilai Jaro Winkler *distance* dengan persamaan (2.2) adalah:

$$d_w = 1 + \left(6.0{,}1(1-1)\right) = 1$$

2. Perhitungan kemiripan kata "gagu"

Tabel 4.14 Perhitungan Kemiripan "gagu"

	g	a	n	g	g	u
g	1	0	0	0	0	0
a	0	1	0	0	0	0
g	0	0	0	1	0	0
u	0	0	0	0	0	1

Dengan:

m = 4

$|s_1|$ = 6

$|s_2|$ = 4

t = 0

l = 2

p = 0.1

Sehingga perhitungan nilai Jaro *distance* dengan persamaan (2.1) adalah:

$$d_j = \frac{1}{3} \times \left(\frac{4}{6} + \frac{4}{4} + \frac{4-0}{4}\right) = 0{,}88$$

Dan nilai Jaro Winkler *distance* dengan persamaan (2.2) adalah:

$$d_w = 1 + \left(2.0{,}1(1-0{,}88)\right) = 0{,}911$$

3. Perhitungan kemiripan kata "gancu"

Tabel 4.15 Perhitungan Kemiripan "gancu"

	g	a	n	g	g	u
g	1	0	0	0	0	0
a	0	1	0	0	0	0
n	0	0	1	0	0	0
c	0	0	0	0	0	0
u	0	0	0	0	0	1

Dengan:

m = 4

$|s_1|$ = 6

$|s_2|$ = 5

t = 0

l = 3

p = 0.1

Sehingga perhitungan nilai Jaro *distance* dengan persamaan (2.1) adalah:

$$d_j = \frac{1}{3} \times \left(\frac{4}{6} + \frac{4}{5} + \frac{4-0}{4}\right) = 0{,}822$$

Dan nilai Jaro Winkler *distance* dengan persamaan (2.2) adalah:

$$d_w = 1 + \left(6.0{,}1(1-1)\right) = 0{,}875$$

4. Perhitungan kemiripan kata “gandu”

Tabel 4.16 Perhitungan Kemiripan “gandu”

	g	a	n	g	g	u
g	1	0	0	0	0	0
a	0	1	0	0	0	0
n	0	0	1	0	0	0
d	0	0	0	0	0	0
u	0	0	0	0	0	1

Dengan:

m = 4

$|s_1|$ = 6

$|s_2|$ = 5

t = 0

l = 3

p = 0.1

Sehingga perhitungan nilai Jaro *distance* dengan persamaan (2.1) adalah:

$$d_j = \frac{1}{3} \times \left(\frac{4}{6} + \frac{4}{5} + \frac{4-0}{4}\right) = 0{,}822$$

Dan nilai Jaro Winkler *distance* dengan persamaan (2.2) adalah:

$$d_w = 1 + \left(6.0{,}1(1-1)\right) = 0{,}875$$

5. Perhitungan kemiripan kata “gagau”

Tabel 4.17 Perhitungan Kemiripan “gagau”

	g	a	n	g	g	u
g	1	0	0	0	0	0
a	0	1	0	0	0	0
g	0	0	0	1	0	0
a	0	0	0	0	0	0
u	0	0	0	0	0	1

Dengan:

m = 4

$|s_1|$ = 6

$|s_2|$ = 5

t = 0

l = 2

p = 0.1

Sehingga perhitungan nilai Jaro *distance* dengan persamaan (2.1) adalah:

$$d_j = \frac{1}{3} \times \left(\frac{4}{6} + \frac{4}{5} + \frac{4-0}{4}\right) = 0{,}822$$

Dan nilai Jaro Winkler *distance* dengan persamaan (2.2) adalah:

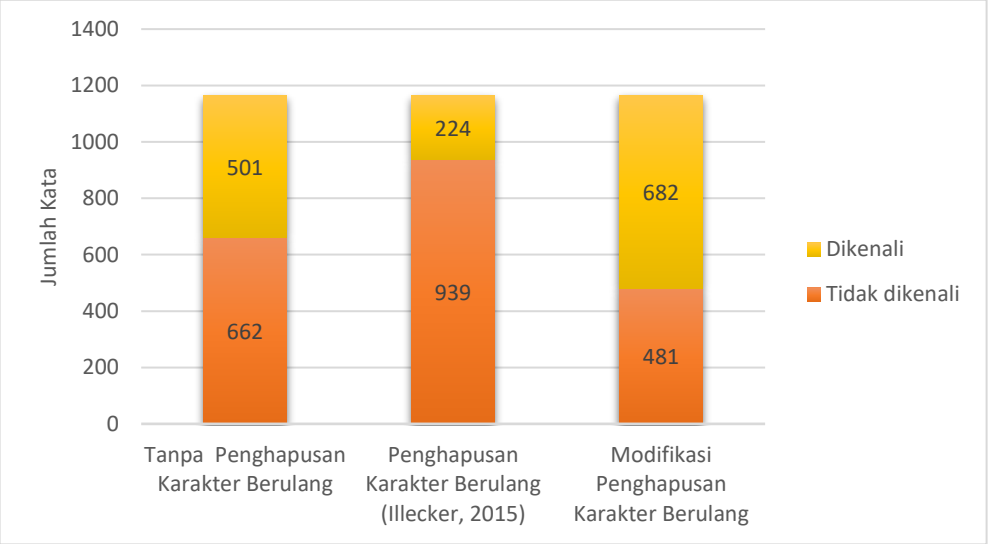
$$d_w = 1 + \left(2.0{,}1(1 - 0{,}822)\right) = 0{,}857$$

Setelah dihitung kemiripan dengan kamus Bahasa Indonesia, kemudian pada tahap lima ditentukan kata kunci dengan mencari nilai bobot yang mendekati dengan 1 sebagai kunci pada tahap berukitnya yaitu kata “ganggu” pada kamus kata baku dengan bobot sebesar 1. Karena bobot yang didapatkan sama dengan 1 sehingga tidak dilanjutkan ke tahap berikutnya yaitu penghapusan kemungkinan penghapusan karakter berulang dan Keluaran yang dihasilkan adalah kata “ganggu”.

Setelah praproses data selesai kemudian dilakukan pengujian signifikansi pada hasil dari praproses. pengujian ini dilakukan untuk mengetahui apakah performa dari praproses yang didapat dengan modifikasi penghapusan karakter berulang berbeda signifikan dengan tidak dilakukannya modifikasi penghapusan

karakter berulang. Pengujian tersebut dilakukan dengan menggunakan uji statistic paired t test. Masukan dari pengujian ini adalah jumlah karakter yang dikenali oleh kamus setelah dilakukannya tahap praproses baik menggunakan modifikasi penghapusan karakter berulang dan tanpa menggunakan modifikasi penghapusan karakter berulang. Pengujian dilakukan dengan perhitungan nilai p-value dengan convidence level 95% seperti yang dilakukan oleh (Bandhakavi et al., 2016; Hutto and Gilbert, 2014). Nilai p-value (sig) dibawah 0,05 menunjukkan adanya perbedaan signifikan antara kedua kelompok data. Dari pengujian yang dilakukan p-value = 0,00000000000000022 dan t = -16.640. Karena nilai p-value kurang dari 0,05 maka disimpulkan data yang diolah tidak menggunakan modifikasi penghapusan karakter berulang lebih rendah sigifikan nilainya dari data yang diolah menggunakan modifikasi penghapusan karakter berulang. Hasil pengujian uji t dapat dilihat pada Lampiran C.

Setelah dilakukan pengujian dengan menggunakan berbagai macam skenario, kemudian dilakukan pengujian dengan menggunakan data twitter. Pengujian dilakukan dengan membandingkan keluaran yang dihasilkan dengan menggunakan tanpa penghapusan karakter berulang, dengan penghapusan karakter berulang, dan dengan modifikasi penghapusan karakter berulang. Dari hasil yang didapatkan seperti yang dijelaskan pada Gambar 4.4, dapat dilihat bahwa proses tanpa menggunakan penghapusan karakter berulang dapat mengenali 43% dari 1163 kata yang mengalami perulangan sedangkan 57% lainnya tidak mampu dikenali. Namun terjadi penurunan jumlah kata yang dapat dikenali ketika menggunakan penghapusan karakter berulang. Sedangkan penghapusan karakter berulang dapat mengenali sebanyak 19%. Hasil terbaik didapatkan dengan menggunakan modifikasi penghapusan karakter berulang dapat mengungguli tanpa penghapusan karakter berulang dan menggunakan penghapusan karakter berulang dengan jumlah kata yang dapat dikenali sebesar 59% dari 1163 kata yang mengalami perulangan.



Gambar 4.4 Perbandingan Kinerja Penghapusan Karakter Berulang

Pada proses tanpa menggunakan penghapusan karakter berulang hanya dapat mengenali 43% dari keseluruhan kata yang mengalami perulangan dikarenakan terdapat kata yang mengalami perulangan karakter sebanyak 662 kata yang mengalami kesalahan dalam perulangan seperti "yaaa" yang seharusnya perulangannya dihapus menjadi "ya" yang menyebabkan kata tersebut tidak dapat dikenali dengan baik. Berikut merupakan contoh daftar kata yang tidak bisa diatasi tanpa menggunakan penghapusan karakter berulang namun bisa diatasi dengan menggunakan modifikasi penghapusan karakter berulang seperti pada Tabel 4.18. Detail kata yang tidak dapat ditangani tanpa penghapusan karakter berulang dapat dilihat pada lampiran A.

Tabel 4.18 Contoh Kata Yang Tidak Dapat Ditangani Tanpa Penghapusan Karakter Berulang

No	Kata	Frekuensi Muncul
1	retweet	33
2	follow	21
3	yaa	21
4	twitter	17
5	speed	17
6	speedy	14
7	error	12
8	off	11
9	ttp	10

Tabel 4.18 Contoh Kata Yang Tidak Dapat Ditangani Tanpa Penghapusan Karakter Berulang (lanjutan)

No	Kata	Frekuensi Muncul
10	responnya	10
11	hallo	9
12	password	9
13	cc	9
14	freedom	9
15	limitless	8
16	connect	7
17	yaaa	7
18	billing	6
19	gaada	6
20	tweet	6

Pada proses menggunakan penghapusan karakter berulang hanya dapat mengenali 19% dari keseluruhan kata yang mengalami perulangan dikarenakan terdapat kata yang mengalami perulangan karakter sebanyak 939 kata yang mengalami kesalahan dalam perulangan seperti “gangguan”, “minggu”, dan “saat” yang seharusnya perulangannya tidak boleh dihapus sehingga menyebabkan kata tersebut tidak dapat dikenali dengan baik. Berikut merupakan contoh daftar kata yang tidak bisa diatasi dengan menggunakan penghapusan karakter berulang namun bisa diatasi dengan menggunakan modifikasi penghapusan karakter berulang seperti pada Tabel 4.19. Detail kata yang tidak dapat ditangani dengan penghapusan karakter berulang dapat dilihat pada lampiran A.

Tabel 4.19 Contoh Kata Yang Tidak Dapat Ditangani Penghapusan Karakter Berulang

No	Kata	Frekuensi Muncul
1	ganguan	33
2	retwet	33
3	maf	32
4	bantuanya	24
5	mingu	24
6	folow	21
7	sat	20
8	pelangan	19
9	nga	18
10	jaringanya	17

Tabel 4.19 Contoh Kata Yang Tidak Dapat Ditangani Penghapusan Karakter Berulang (lanjutan)

No	Kata	Frekuensi Muncul
11	twiter	17
12	sped	17
13	mengunakan	16
14	spedy	14
15	ngak	13
16	of	11
17	nungu	10
18	enga	10
19	responya	10
20	tengang	9

Pada modifikasi penghapusan karakter berulang yang dilakukan dalam penelitian ini hanya dapat mengenali 59% dari keseluruhan kata yang mengalami perulangan. Hal itu dikarenakan 41% diantaranya adalah kata memiliki perulangan yang tidak bisa dikenali dikarenakan kata tersebut mengalami perulangan karakter Bahasa inggris, singkatan, kata yang tidak sesuai dengan EYD (Ejaan Yang Disempurnakan) sehingga proses pencarian kesamaan kata dengan kamus Bahasa Indonesia tidak berjalan dengan baik seperti pada Tabel 4.20. Detail kata yang tidak dapat ditangani dengan modifikasi penghapusan karakter berulang dapat dilihat pada lampiran A.

Tabel 4.20 Contoh Kata Yang Tidak Dapat Ditangani Modifikasi Penghapusan Karakter Berulang

No	Kata	Frekuensi Muncul
1	retweet	33
2	follow	21
3	twitter	17
4	speed	17
5	speedy	14
6	off	11
7	responnya	10
8	password	9
9	cc	9
10	freedom	9
11	limitless	8

Tabel 4.20 Contoh Kata Yang Tidak Dapat Ditangani Modifikasi Penghapusan Karakter Berulang (lanjutan)

No	Kata	Frekuensi Muncul
12	connect	7
13	billing	6
14	gaada	6
15	tweet	6
16	bb	6
17	facebook	6
18	useetv	6
19	ooredoo	6
20	kk	5

Dari uji coba yang dilakukan dengan membandingkan hasil dari beberapa jenis pengujian dapat diketahui modifikai yang dilakukan menghasilkan performa klasifikasi terbaik seperti yang dapat dilihat pada Tabel 4.21. Jenis pengujian ini menghasilkan akurasi, presisi, recall, dan f-measure sebesar 72.71%, 73.2%, 72.2%, dan 72.9%. Sedangkan hasil paling rendah didapatkan menggunakan jaccard dengan akurasi, presisi, recall, dan f-measure sebesar 67.54%, 67.9%, 67.5%, dan 67.7%. Rendahnya nilai yang diperoleh menggunakan jaccard dikarenakan terjadi kesalahan dalam proses pencarian kemiripan dengan kamus yang menyebabkan kata memiliki makna yang berbeda. Kesalahan dalam pemrosesan seperti kata “kecepatannya” menjadi “kedatangnnya” yang disebabkan kata “kedatangnnya” memiliki bobot paling tinggi dari lainnya yang mengakibatkan kata “kedatangnnya” dipilih sebagai keluaran pada tahap ini.

Tabel 4.21 Perbandingan Akurasi Penggunaan Tahap Penghapusan Karater Berulang

Jenis Pengujian	Akurasi (%)	Presisi (%)	*Recall* (%)	F-measure (%)
Tanpa penghapusan karakter berulang	70.67	71	70.7	70.8
Metode Illecker (Illecker, 2015)	71.25	72.1	71.3	71.6
Metode Jaccard (Choi et al., 2014)	67.54	67.9	67.5	67.7
Modifikasi penghapusan katakter berulang	72.71	73.2	72.7	72.9

4.5.2. Hasil dan Analisis Uji Coba Perbandingan Penghapusan Kata Henti

Uji coba ini dilakukan dengan membandingkan hasil klasifikasi menggunakan dan tanpa menggunakan penghapusan kata henti. Pengujian ini dilakukan untuk mengetahui dampak dari terhapusnya kata mengandung sentimen yang terhapus oleh tahap penghapusan kata henti seperti pada Tabel 4.22. Kata tersebut dihapus karena terdaftar pada *stop list* sehingga mengakibatkan perbaikan pada tahap sebelumnya yaitu penghapusan karakter berulang tidak dapat berjalan maksimal.

Tabel 4.22 Kata Yang Berhasil Diperbaiki Namun Terhapus Penghapusan Kata Henti

No	Kata	Frekuensi Muncul
1	guna	20
2	saat	19
3	jawab	6
4	baik	4
5	tanya	4
6	apa	3
7	hingga	3
8	bulan	2
9	bisa	2
10	pasti	2
11	lain	2
12	bukan	2
13	sehingga	2
14	siap	2
15	lanjut	2
16	ikut	2
17	sudah	1
18	belum	1
19	tidak	1
20	berapa	1
21	balik	1
22	saya	1
23	lama	1

Tabel 4.22 Kata Yang Berhasil Diperbaiki Namun Terhapus Penghapusan Kata Henti (lanjutan)

No	Kata	Frekuensi Muncul
24	minta	1
25	seperti	1
26	umum	1
27	lagi	1
28	ada	1
29	tinggi	1
30	juga	1
Total		**86**

Hasil yang didapat dari pengujian dapat dilihat pada Tabel 4.23. Hasil akurasi terbaik didapatkan dengan tanpa penggunaan tahap penghapusan kata henti dan hasil terendah dengan penghapusan kata henti sebesar 74.46% dan 72.71%. Dari hasil yang diperoleh dapat diketahui bahwa kata yang mengandung sentimen sangat mempengaruhi hasil dari kinerja klasifikasi karena pengukuran yang dilakukan bergantung pada sentimen yang terkandung didalam *tweet*. Jika kata yang mengandung sentimen terhapus maka akan berdampak pada akurasi pada klasifikasi. Hal itu dibuktikan dengan meningkatnya nilai akurasi ketika tidak menggunakan tahap penghapusan kata henti.

Tabel 4.23 Perbandingan Akurasi Penggunaan Tahap Penghapusan Kata Henti

Jenis Pengujian	Akurasi (%)	Presisi (%)	*Recall* (%)	F-measure (%)
Dengan penghapusan kata henti	72.71	73.2	72.7	72.9
Tanpa penghapusan kata henti	**74.46**	**74.9**	**74.5**	**74.6**

Terhapusnya kata tersebut menyebabkan hasil dari modifikasi penghapusan karakter berulang tidak berjalan maksimal. Hal itu dikarenakan pengukuran yang dilakukan pada penelitian ini bergantung pada sentimen yang terkandung didalam tweet. Jika sentimen yang terdaptar dalam *stop list* dihapus maka akan

mempengaruhi hasil dari proses berikutnya. Oleh sebab itu tahap penghapusan kata henti tidak digunakan pada penelitian ini.

4.5.3. Hasil dan Analisis Uji Coba Praproses

Uji coba ini dilakukan berbagai jenis pengujian dengan berbagai tahapan pada praproses yang dapat dilihat pada Tabel 4.24. Hasil dari pengujian ini dapat diketahui modifikasi penghapusan kata henti mampu menghasilkan nilai akurasi terbaik dengan nilai akurasi, presisi, recall, dan f-measure sebesar 71.71%, 73.2%, 72.7%, dan 72.9%. Sedangkan hasil paling rendah didapatkan tanpa menggunakan praproses dengan nilai akurasi, presisi, recall, dan f-measure sebesar 64.96%, 64.8%, 65%, dan 64.9%.

Tabel 4.24 Hasil Pengujian Tahapan Praproses

Jenis Pengujian	Akurasi (%)	Presisi (%)	*Recall* (%)	F-measure (%)
Tanpa praproses	64.96	64.8	65	64.9
Tanpa penghapusan karakter berulang	70.67	71	70.7	70.8
Tanpa stemming	70.08	70.5	70.1	70.3
Tanpa konversi kata tidak baku	69.88	70.4	69.9	70.1
Keseluruhan tahap praproses dengan metode llecker	71.71	72.2	71.7	71.9
Keseluruhan tahap praproses dengan metode Jaccard	68.04	68.4	68	68.2
Keseluruhan tahap praproses dengan Modifikasi	**74.46**	**74.9**	**74.5**	**74.6**

Dari hasil pengujian ini diketahui bahwa keseluruhan tahap praproses mampu menghasilkan performa tertinggi. Kombinasi tahap ini menghasilkan nilai akurasi, presisi, recall, dan f-measure sebesar 74.46%, 74.9%, 74.5%, dan 74.6%. Sedangkan hasil paling rendah didapatkan jika tidak menggunakan praproses dengan nilai akurasi, presisi, recall, dan f-measure sebesar 64.96%, 64.8%, 65%, dan 64.9%.

Performa terendah didapatkan jika tidak menggunakan tahap konversi kata tidak baku dengan nilai akurasi, presisi, recall, dan f-measure sebesar 69.88%, 70.4%, 69.9%, dan 70.1%. Hal ini disebabkan karena penggunaan bahasa yang tidak sesuai dengan EYD dan mengalami singkatan sering muncul pada data twitter sehingga tahap ini memiliki pengaruh signifikan dalam memperbaiki kata. Performa terendah berikutnya adalah jika menggunakan *jaccard* dengan nilai akurasi, presisi, recall, dan f-measure sebesar 68.04%, 68.4%, 68%, dan 68.2%. Rendahnya nilai yang diperoleh menggunakan *jaccard* dikarenakan terjadi kesalahan dalam proses pencarian kemiripan dengan kamus yang menyebabkan kata memiliki makna yang berbeda. Kesalahan dalam pemrosesan seperti kata "kecepatannya" menjadi "kedatangnnya" yang disebabkan kata "kedatangnnya" memiliki bobot paling tinggi dari lainnya yang mengakibatkan kata "kedatangnnya" dipilih sebagai keluaran pada tahap ini. Performa terendah berikutnya adalah jika tidak menggunakan tahap stemming dengan nilai akurasi, presisi, recall, dan f-measure sebesar 70.08%, 70.5%, 70.1%, dan 70.3%. Tidak terpaut jauh dari tahap stemming, tahap tanpa penghapusan karakter berulang dapat menghasilkan performa klasifikasi dengan nilai akurasi, presisi, recall, dan f-measure sebesar 70.67%, 71%, 70.7%, dan 70.8%. Proses dengan menggunakan penghapusan karakter berulang dapat mengungguli dari *jaccard* dengan menghasilkan performa klasifikasi dengan nilai akurasi, presisi, recall, dan f-measure sebesar 71.71%, 72.2%, 71.7%, dan 71.9%. Hal ini dikarenakan terdapat perulangan karakter yang tidak dapat diproses dengan baik seperti "gangguan", "minggu", dan "saat" yang seharusnya perulangannya tidak boleh dihapus sehingga menyebabkan kata tersebut tidak dapat dikenali dengan baik.

Hasil pengujian menunjukkan peningkatan kinerja ketika menggunakan modifikasi penghapusan karakter berulang, baik dari akurasi maupun dari kata yang dapat dikenali. Hal ini dapat disimpulkan bahwa modifikasi tersebut memiliki peran yang signifikan dari aspek kesalahan makna kata dan memiliki peran yang cukup signifikan dalam praproses pada penelitian ini. Modifikasi penghapusan karakter dengan mengenali kata sebesar 59%. Hal itu disebabkan karena terjadi perulangan karakter, perulangan yang mengandung penggunaan bahasa asing, penggunaan singkatan, dan kata yang tidak sesuai EYD sehingga modifikasi penghapusan

karakter tidak dapat menangani seluruh kata. Modifikasi dilakukan untuk memproses kata yang memiliki akhiran sama dengan karakter awal imbuhan seperti kata “pelanggannya”, terdiri dari kata “pelanggan” kemudian mendapat imbuhan “nya”. Jika tidak dilakukan modifikasi penghapusan karakter, maka kata “pelanggannya” akan berubah menjadi “pelanganya”. Hal ini menyebabkan kata tidak dapat diproses oleh *stemmer*. Selain itu modifikasi juga dapat meningkatkan kinerja pada tahap *stemming* dan penghapusan kata henti. Peningkatan kinerja *stemming* dibuktikan pada Gambar 4.4, modifikasi penghapusan karakter berulang dapat mengenali 682 kata, sedangkan penghapusan karakter berulang dapat mengenali 224 kata dari total keseluruhan 1163 kata yang mengalami perulangan. Di sisi lain peningkatan kinerja penghapusan kata henti dibuktikan pada Tabel 4.22. Sebanyak 86 kata dapat direduksi oleh penghapusan kata henti ketika menggunakan modifikasi penghapusan karakter berulang, sehingga menurunkan tingkat keberagaman kata yang memiliki arti yang sama.

4.6. Kontribusi Penelitian

Pada sub-bab ini akan dibahas kontribusi dari penelitian yang telah dilakukan. Kontribusi yang disajikan meliputi kontribusi keilmuan dan kontribusi praktis.

4.6.1. Kontribusi Keilmuan

Kontribusi keilmuan didapatkan setelah dilakukannya tahap pengujian dan analisis. Metode *illecker* pada penelitian yang dilakukan oleh (Illecker, 2015) masih memiliki beberapa kelemahan. Hal itu dapat disebabkan karena metode ini melakukan penghapusan dengan menghilangkan karakter yang mengalami pengulangan. Permasalahan terjadi ketika memproses kata yang memiliki perulangan pada kata bakunya seperti “sehingga”, “tunggu”, dan “saat”. Metode ini akan menghapus perulangan “sehingga” menjadi “sehinga”, “tunggu” menjadi “tungu”, dan “saat” menjadi “sat” sehingga kata akan kehilangan makna dan tidak dapat diproses dengan baik pada tahap berikutnya. Selain itu permasalahan terjadi ketika kata mendapatkan imbuhan seperti kata “pelanggannya”, yang terdiri dari kata “pelanggan” kemudian mendapatkan imbuhan “nya” yang mengakibatkan kata mengalami perulangan karakter “g” dan “n” sehingga tidak dapat dikenali oleh kamus Bahasa

Indonesia. Jika kata tersebut diproses menggunakan metode ini, karakter “g” dan “n” akan direduksi menjadi “pelanganya” yang mengakibatkan kata tidak dapat dikenali meskipun imbuhan “nya” telah dihapus.

Selain itu metode *jaccard* pada penelitian yang dilakukan oleh (Choi et al., 2014) dengan melakukan perhitungan *smilarity* pada kamus untuk memperbaiki kesalahan penulisan perulangan dengan cara menemukan kata yang memiliki kesamaan terdekat nampaknya juga tidak bisa menangani permasalahan perulangan kata. Hasil yang diperoleh dengan metodi ini terbilang rendah. Rendahnya nilai yang diperoleh menggunakan jaccard dikarenakan terjadi kesalahan dalam proses pencarian kemiripan dengan kamus yang menyebabkan kata memiliki makna yang berbeda. Kesalahan dalam pemprosesan pencarian kemiripan kata “kecepatannya” menjadi “kedatangnnya” yang disebabkan kata “kedatangnnya” memiliki bobot paling tinggi dari kata lainnya yang mengakibatkan kata “kedatangnnya” dipilih sebagai keluaran pada tahap ini.

Oleh sebab itu pada penelitian ini ditambahkan proses untuk memeriksa apakah didalam kata tersebut mengandung perulangan karakter sejenis lebih dari satu perulangan atau tidak. Proses ini dilakukan untuk menghindari reduksi perulangan yang berlebihan yang akan membuat kata kehilangan makna. Jika tahap ini tidak dilakukan maka kata seperti “sehingga”, perulangan karakter “g” akan direduksi menjadi karakter tunggal yang menyebabkan kata “sehinga” tidak dapat diproses dengan baik karena kehilangan makna.

Selain itu digunakan pula algoritma *jaro winkler* untuk mencari kemiripan kata dengan kamus. Pencarian ini dilakukan untuk mencari kata yang memiliki bobot paling mendekati nilai satu yang menandakan bahwa ditemukannya kata baku yang kemiripannya 100%. Namun jika bobot tertinggi yang dihasilkan kurang dari 1, maka kata dengan bobot tertinggi akan dipilih untuk mengurangi keanekaragaman pada kata yang sama.

Kemudian ketiga metode tersebut diuji dengan data dan tahapan praproses yang sama dan hasilnya seperti berikut:

Tabel 4.25 Perbandingan Akurasi pada Praproses

Jenis Pengujian	Akurasi (%)
Keseluruhan tahap praproses dengan metode llecker	71.71
Keseluruhan tahap praproses dengan metode Jaccard	68.04
Keseluruhan tahap praproses dengan Modifikasi	**74.46**

Dari hasil pegujian pada Tabel 4.25 dapat disimpulkan bahwa keakuratan meningkat ketika dilakukan modifikasi. Peningkatan akurasi didapatkan karena proses pemeriksaan perulangan dapat membedakan kata dari masing-masing jenis perulangan. Hal itu membuat kata dapat diperlakukan sesuai dengan jenis perulangan yang terjadi sehingga tidak salah dalam melakukan penghapusan karakter.

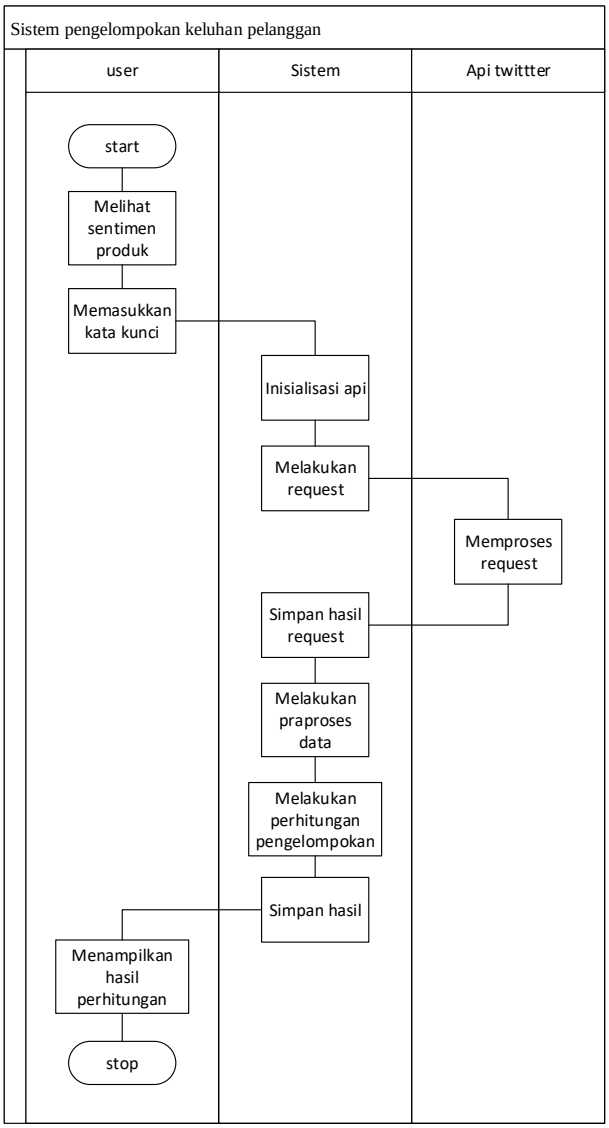
Selain itu didapatkan pula jenis perulangan pada kata yang terkandung pada teks berbahasa Indonesia. Jika dikelompokkan dapat dibedakan menjadi empat macam diantaranya adalah 1) kata baku mengandung perulangan yang mengalami perulangan karakter lebih dari satu jenis karakter seperti “pelanggannya”, “mengganggu”; 2) Kata baku mengandung perulangan yang tidak mengalami perulangan karakter seperti “maaf”, “manfaat”; 3) kata baku tidak mengandung perulangan yang mengalami perulangan karakter seperti “kecewaaa”, “lagiii”; 4) kata baku tidak mengandung perulangan yang mengalami perulangan karakter lebih dari satu jenis karakter seperti “pertanyaannya”, “masiiihhh”. Dari empat jenis perulangan yang telah dijelaskan memerlukan penanganan yang berbeda untuk mendapatkan kata yang sesuai dengan kata bakunya.

4.6.2. Kontribusi Praktis

Kontribusi praktis dari penelitian ini adalah penggunaan metode modifikasi penghapusan karakter berulang dapat diterapkan pada praproses dalam melakukan pengelompokan data keluhan layanan telekomunikasi seluler yang sumber datanya diambil melalui media sosial dimana sering sekali terjadi kesalahan dalam penulisan.

Berdasarkan hasil pengujian, disarankan untuk menggunakan tahapan modifikasi penghapusan karakter berulang untuk menangani kesalahan pengulangan dalam kata. Berdasarkan gambar 4.4, modifikasi yang dilakukan dapat mengungguli metode lainnya dengan mampu menangani sebanyak 59% dari kata yang memiliki perulangan karakter tanpa membuat berubahnya makna pada kata.

Selain kontribusi dari hasil pengujian dan analisis data yang telah dilakukan, juga dirancang sistem pengelompokan keluhan pelanggan pada penyedian layanan telekomunikasi untuk melakukan analisis sentimen pengguna terhadap produk dan layanan yang mereka berikan. Perancangan alur sistem pengelompokan dijelaskan dalam bentuk *flowchart* pada Gabar 4.5.



Gambar 4.5 *Flowchart* Sistem Pengelompokan Keluhan Pelanggan

Proses pengelompokan dimulai dengan user memasukkan kata kunci berupa akun twetter yang digunakan untuk kata kunci pencarian yang dilakukan pada tahap berikutnya. Setelah user memasukkan kata kunci kemudian sistem akan melakukan inisialisasi api berupa verifikasi token yang telah didaftarkan pada *Twitter Developers*. Jika token terdaftar makan proses request akan dijalankan oleh api twitter dengan mengirimkan hasil request berupa JSON yang akan diterima oleh sistem. Setelah JSON diterima kemudian hasil yang didapatkan akan disimpan kedalam *data base* sebagai persiapan dilakukannya proses perhitungan pengelompokan sentiment. Setelah data didapat kemudian sistem mealakukan praproses seperti tokenizing, pembersihan derau, case folding, penghapusan karakter berulang, stemming, dan konversi kata tidak baku. Setelah data bersih dari derau, kemudian dilakukan perhitungan pengelompokan berdasarkan kelasnya. Dari hasil perhitungan kemudian disimpan dan ditampilkan kepada user untuk dilakukan analisis lebih dalam.

Halaman ini sengaja dikosongkan

BAB V
KESIMPULAN DAN SARAN

Dalam bab ini dijelaskan kesimpulan dari hasil uji coba dan analisis hasil yang dilakukan sesuai dengan skenario uji coba.

5.1. Kesimpulan

Setelah penelitian mengenai perancangan metode penghapusan karakter berulang yang telah selesai dilaksanakan, maka berikut ini merupakan kesimpulan dari penelitian ini.

a. Salah satu permasalahan pada data twitter jika digunakan dalam klasifikasi sentimen adalah bahasa yang digunakan tidak resmi seperti penggunaan perulangan karakter yang berulang yang menyebabkan kata tidak dapat dikenali dengan baik pada tahap praproses. Hasil penelitian ini menunjukan bahwa modifikasi penghapusan karakter berulang yang dikembangkan dapat menangani kata mengandung perulangan yang tidak dapat ditangani jika menggunakan penghapusan karakter berulang. Pada beberapa pengujian yang dilakukan dapat dilihat bahwa proses tanpa menggunakan penghapusan karakter berulang dapat mengenali kata yang mengalami perulangan sebanyak 43% kata, sedangkan penghapusan karakter berulang dapat mengenali kata yang mengalami perulangan sebanyak 19% kata, hasil terbaik didapatkan dengan menggunakan modifikasi penghapusan karakter berulang dapat mengenali kata yang mengalami perulangan sebanyak 59% kata.
b. Dari hasil pengujian dengan membandingkan tanpa, menggunakan, dan modifikasi penghapusan karakter berulang diketahui modifikasi yang dilakukan dapat menghasilkan performa klasifikasi paling baik dengan nilai akurasi, presisi, recall, dan f-measures sebesar 72.71%, 73.2%; 72.7%; dan 72.9%. Sedangkan jika tidak dilakukan modifikasi menghasilkan performa klasifikasi paling baik dengan nilai akurasi, presisi, recall, dan f-measures sebesar 71.25%; 72.1%; 71.3%; dan 71.6%.
c. Modifikasi yang dilakukan memiliki peran yang signifikan dari aspek kesalahan makna dari kata, hasil terbaik didapatkan dengan menggunakan modifikasi penghapusan karakter dengan kata yang berhasil dikenali sebesar

59%. Hal itu disebabkan karena perulangan yang terjadi pada data tidak hanya karena perulangan karakter, namun juga mengandung penggunaan bahasa selain Bahasa Indonesia, penggunaan singkatan, dan kata yang tidak sesuai dengan EYD sehingga modifikasi penghapusan karakter tidak dapat menangani seluruh kata.

d. Selain itu modifikasi yang dilakukan dapat meningkatkan kinerja pada tahap *stemming* dan penghapusan kata henti. Peningkatan kinerja *stemming* dibuktikan dengan jumlah kata yang dapat dikenali dengan modifikasi penghapusan karakter berulang sebesar 682 kata sedangkan penghapusan karakter berulang sebesar 224 kata dari total keseluruhan 1163 kata yang mengalami perulangan. Di sisi lain peningkatan kinerja penghapusan kata henti dibuktikan dengan terdapat 613 kata yang dapat direduksi oleh penghapusan kata henti ketika menggunakan modifikasi penghapusan karakter berulang sehingga dapat menurunkan tingkat keberagaman kata yang memiliki arti dan maksud yang sama.

5.2. Saran

Bahasa yang digunakan pada data twitter yang digunakan dalam penelitian ini adalah Bahasa indoesia. Tetapi dalam kenyataannya penggunaan Bahasa inggris masi ditemukan pada data. Dari hasil pengujian ditemukan kata berbahasa inggris yang tidak mampu dikenali karena kamus yang digunakan adalah kamus Bahasa Indonesia. Pada penelitian yang lebih lanjut dapat menerapkan teknik untuk mengenali dua bahasa (Inggris dan Indonesia) untuk menghasilkan hasil praproses yang lebih baik.

DAFTAR PUSTAKA

Akaichi, J., 2013. Social Networks' Facebook' Statutes Updates Mining for Sentiment Classification, in: 2013 International Conference on Social Computing. Presented at the 2013 International Conference on Social Computing, pp. 886–891.

Amarouche, K., Benbrahim, H., Kassou, I., 2015. Product Opinion Mining for Competitive Intelligence. Procedia Comput. Sci., International Conference on Advanced Wireless Information and Communication Technologies (AWICT 2015) 73, 358–365. doi:10.1016/j.procs.2015.12.004

Amolik, A., Jivane, N., Bhandari, M., Venkatesan, M., 2016. Twitter Sentiment Analysis of Movie Reviews using Machine Learning Techniques.

Aprilla, D., Baskoro, D.A., Ambarwati, L., Wicaksana, I.W.S., 2013. Belajar Data Mining Dengan Rapid Miner.

Arifiyanti, A.A., 2015. Ekstrasi Fitur Pada Konten Jejaring Sosial Twitter Berbahasa Indonesia Dalam Peningkatan Kinerja Klasifikasi Sentimen. Intitut Teknologi Sepuluh Nopember Surabaya.

Aurangzeb, K., Baharum, B., Lam, H., Khairullah, khan, 2010. A Review of Machine Learning Algorithms for Text-Documents Classification.

Bahrainian, S.-A., Dengel, A., 2013. Sentiment Analysis and Summarization of Twitter Data. 2013 IEEE 16th Int. Conf. Comput. Sci. Eng.

Bandhakavi, A., Wiratunga, N., Massie, S., Deepak, P., 2016. Emotion-Corpus Guided Lexicons for Sentiment Analysis on Twitter. Research and Development in Intelligent Systems XXXIII, DOI 10.1007/978-3-319-47175-4_5.

Basari, A.S.H., Hussin, B., Ananta, I.G.P., Zeniarja, J., 2013. Opinion Mining of Movie Review using Hybrid Method of Support Vector Machine and Particle Swarm Optimization. Procedia Eng., Malaysian Technical Universities Conference on Engineering & Technology 2012, MUCET 2012 53, 453–462.

Bhuta, S., Doshi, U., 2014. A review of techniques for sentiment analysis Of Twitter data, in: 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT). Presented at the 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), pp. 583–591.

Bouazizi, M., Ohtsuki, T., 2015. Opinion mining in Twitter: How to make use of sarcasm to enhance sentiment analysis. Presented at the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 1594–1597.

Choi, D., Kim, J., Kim, P., 2014. A Method for Normalizing Non-standard Words in Online Social Network Services: A Case Study on Twitter. Springer International Publishing Switzerland 2014, DOI: 10.1007/978-3-319-05939-6 35.

Clark, E., Araki, K., 2011. Text Normalization in Social Media: Progress, Problems and Applications for a Pre-Processing System of Casual English. Procedia - Soc. Behav. Sci., Computational Linguistics and Related Fields 27, 2–11. doi:10.1016/j.sbspro.2011.10.577

Cohen, W.W., Ravikumar, P., Fienberg, S.E., 2003. A Comparison of String Distance Metrics for Name-Matching Tasks.

Coutinho, D.P., Figueiredo, M. a. T., 2013. An information theoretic approach to text sentiment analysis. ResearchGate 577–580.

Damar, A.M., 2016. 3 Fakta Mengejutkan Pengguna Internet di Indonesia [WWW Document]. URL http://tekno.liputan6.com/read/2435997/3-fakta-mengejutkan-pengguna-internet-di-indonesia (accessed 5.11.16).

Daniel, M., Neves, R.F., Horta, N., 2017. Company event popularity for financial markets using Twitter and sentiment analysis. Expert Syst. Appl. 71, 111–124.

Dreßler, K., Ngomo, A.-C.N., 2014. Time-Efficient Execution of Bounded Jaro-Winkler Distances.

Falcon Design Studio, 2012. Social Media's Positive Impact for Most Businesses [WWW Document]. URL http://www.falcondesignstudio.com/social-medias-positive-impact-for-most-businesses/ (accessed 1.6.17).

Garg, Y., 2014. yogeshg/Twitter-Sentiment [WWW Document]. GitHub. URL https://github.com/yogeshg/Twitter-Sentiment (accessed 1.21.17).

Gokulakrishnan, B., Priyanthan, P., Ragavan, T., Prasath, N., Perera, A., 2012. Opinion mining and sentiment analysis on a Twitter data stream, in: International Conference on Advances in ICT for Emerging Regions (ICTer2012). Presented at the International Conference on Advances in ICT for Emerging Regions (ICTer2012), pp. 182–188.

Haddi, E., Liu, X., Shi, Y., 2013. The Role of Text Pre-processing in Sentiment Analysis. Procedia Comput. Sci., First International Conference on Information Technology and Quantitative Management 17, 26–32.

Han, J., Kamber, M., 2000. Data Mining Concept and Techniques, Second edition. ed.

Hsu, C.-W., Chang, C.-C., Lin, C.-J., 2016. A Practical Guide to Support Vector Classification.

Hutto, C.., Gilbert, E., 2014. VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media.

Illecker, M., 2015. Real-time Twitter Sentiment Classification based on Apache Storm.

internetlivestat.com, 2017. Twitter Usage Statistics [WWW Document]. URL http://www.internetlivestats.com/twitter-statistics/ (accessed 1.10.17).

Joachims, T., 2005. Text Categorization with Support Vector Machines: Learning with Many Relevant Features.

Keretna, S., Hossny, A., Creighton, D., 2013. Recognize User Identity in Twitter Social Networks via Text Mining.

Khan, A.U.R., Khan, M., Khan, M.B., 2016. Naïve Multi-label Classification of YouTube Comments Using Comparative Opinion Mining. Procedia Comput. Sci., 4th Symposium on Data Mining Applications, SDMA2016, 30 March 2016, Riyadh, Saudi Arabia 82, 57–64. doi:10.1016/j.procs.2016.04.009

Sastrawi, n.d. Stemming Bahasa Indonesia [WWW Document]. GitHub. URL https://github.com/sastrawi/sastrawi (accessed 1.17.17).

Shirbhate, A.G., Deshmukh, S.N., 2016. Feature Extraction for Sentiment Classification on Twitter Data [WWW Document]. URL https://www.ijsr.net/archive/v5i2/NOV161677.pdf (accessed 1.11.17).

Sumathi, V.., Kousalya, K., Kalaiselvi, R., 2015. A Comparative study on Syntax Matching Algorithms in Semantic Web [WWW Document]. URL http://www.wseas.org/multimedia/journals/computers/2015/a965705-699.pdf (accessed 3.24.17).

Tala, F.Z., 2003. A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia. Universiteit van Amsterdam The Netherlands, Amsterdam.

Vidya, N.A., Fanany, M.I., Budi, I., 2015. Twitter Sentiment to Analyze Net Brand Reputation of Mobile Phone Providers. Procedia Comput. Sci., The Third Information Systems International Conference 2015 72, 519–526.

LAMPIRAN A

Berikut merupakan 100 kata memiliki frekuensi kemunculan tertinggi yang tidak dapat diatasi tanpa penghapusan karakter berulang seperti pada Tabel A.1.

Tabel A.1 Kata yang tidak dapat diatasi tanpa penghapusan karakter berulang

No	Kata	Frekuensi Muncul
1	retweet	33
2	follow	21
3	yaa	21
4	twitter	17
5	speed	17
6	speedy	14
7	error	12
8	off	11
9	ttp	10
10	responnya	10
11	hallo	9
12	password	9
13	cc	9
14	freedom	9
15	limitless	8
16	connect	7
17	yaaa	7
18	billing	6
19	gaada	6
20	tweet	6
21	facebook	6
22	useetv	6
23	ooredoo	6
24	kk	5
25	gallery	5
26	bb	5
27	call	5
28	bbrp	5
29	booster	4
30	free	4
31	connection	4
32	connex	4
33	redeem	4

Tabel A.1 Kata yang tidak dapat diatasi tanpa penghapusan karakter berulang (lanjutan)

No	Kata	Frekuensi Muncul
34	setting	4
35	pass	4
36	channel	4
37	bbm	3
38	speednya	3
39	wifiid	3
40	supportnya	3
41	acces	3
42	offer	3
43	dll	3
44	messagepesan	3
45	iyaa	3
46	ssh	3
47	halooo	2
48	mall	2
49	ehh	2
50	yahh	2
51	support	2
52	jazakumullah	2
53	ussd	2
54	apps	2
55	bella	2
56	bosster	2
57	ww	2
58	app	2
59	smartfreen	2
60	message	2
61	okee	2
62	boongan	2
63	google	2
64	add	2
65	matiin	2
66	penanganannya	2
67	good	2
68	followup	2
69	uufef	2
70	access	2
71	reboot	2

Tabel A.1 Kata yang tidak dapat diatasi tanpa penghapusan karakter berulang (lanjutan)

No	Kata	Frekuensi Muncul
72	makassar	2
73	followback	2
74	speedtest	2
75	ss	2
76	yukk	2
77	broo	2
78	xxl	2
79	handsfree	2
80	tweepscare	1
81	ttdftar	1
82	klikkrm	1
83	dibelaain	1
84	follback	1
85	account	1
86	masaaktif	1
87	sudaah	1
88	bisaa	1
89	bantuannnya	1
90	blackberry	1
91	menggangudg	1
92	yummy	1
93	sooooo	1
94	hhh	1
95	mybankbiixltunai	1
96	beluum	1
97	tidaak	1
98	boosternya	1
99	bii	1
100	terrulus	1

Berikut merupakan 100 kata memiliki frekuensi kemunculan tertinggi yang tidak dapat diatasi penghapusan karakter berulang seperti pada Tabel A.2.

Tabel A.2 Kata yang tidak dapat diatasi penghapusan karakter berulang

No	Kata	Frekuensi Muncul
1	ganguan	33
2	retwet	33

Tabel A.2 Kata yang tidak dapat diatasi penghapusan karakter berulang (lanjutan)

No	Kata	Frekuensi Muncul
3	maf	32
4	bantuanya	24
5	mingu	24
6	folow	21
7	sat	20
8	pelangan	19
9	nga	18
10	jaringanya	17
11	twiter	17
12	sped	17
13	mengunakan	16
14	spedy	14
15	ngak	13
16	of	11
17	nungu	10
18	enga	10
19	responya	10
20	tengang	9
21	semingu	9
22	pasword	9
23	c	9
24	tungu	9
25	oredo	9
26	fredom	9
27	berlanganan	8
28	penguna	8
29	limitles	8
30	tangal	7
31	conect	7
32	langanan	7
33	tingal	7
34	boster	6
35	biling	6
36	gada	6
37	menungu	6
38	twet	6
39	ditangapi	6
40	b	6
41	facebok	6

Tabel A.2 Kata yang tidak dapat diatasi penghapusan karakter berulang (lanjutan)

No	Kata	Frekuensi Muncul
42	usetv	6
43	jawabanya	6
44	perbaikanya	5
45	alhamdulilah	5
46	k	5
47	aces	5
48	chanel	5
49	cal	5
50	fre	4
51	conection	4
52	spednya	4
53	conex	4
54	redem	4
55	seting	4
56	disi	4
57	ditungu	4
58	tangapan	4
59	tulisanya	3
60	bm	3
61	minguan	3
62	pelayananya	3
63	hinga	3
64	pertanyan	3
65	wifid	3
66	suportnya	3
67	smartfren	3
68	kecepatanya	3
69	tagihanya	3
70	sehinga	3
71	sunguh	3
72	ofer	3
73	kelanjutanya	3
74	mesagepesan	3
75	sh	3
76	h	2
77	dinfokan	2
78	pembelianya	2
79	mal	2
80	eh	2
81	suport	2

Tabel A.2 Kata yang tidak dapat diatasi penghapusan karakter berulang (lanjutan)

No	Kata	Frekuensi Muncul
82	jazakumulah	2
83	usd	2
84	aps	2
85	tergangu	2
86	slu	2
87	ap	2
88	mesage	2
89	bongan	2
90	gogle	2
91	screnshot	2
92	alasanya	2
93	ad	2
94	penangananya	2
95	layananya	2
96	mengangu	2
97	kendaran	2
98	god	2
99	folowup	2
100	ufef	2

Berikut merupakan 100 kata memiliki frekuensi kemunculan tertinggi kata yang tidak dapat ditangani modifikasi penghapusan karakter berulang seperti pada Tabel A.3.

Tabel A.3 Kata yang yang tidak dapat ditangani modifikasi penghapusan karakter berulang

No	Kata	Frekuensi Muncul
1	retweet	33
2	follow	21
3	twitter	17
4	speed	17
5	speedy	14
6	off	11
7	responnya	10
8	password	9
9	cc	9
10	freedom	9
11	limitless	8

Tabel A.3 Kata yang yang tidak dapat ditangani modifikasi penghapusan karakter berulang (lanjutan)

No	Kata	Frekuensi Muncul
12	connect	7
13	billing	6
14	gaada	6
15	tweet	6
16	bb	6
17	facebook	6
18	useetv	6
19	ooredoo	6
20	kk	5
21	call	5
22	booster	4
23	free	4
24	connection	4
25	connex	4
26	redeem	4
27	setting	4
28	channel	4
29	bbm	3
30	speednya	3
31	wifiid	3
32	supportnya	3
33	acces	3
34	offer	3
35	messagepesan	3
36	ssh	3
37	hh	2
38	mall	2
39	ehh	2
40	support	2
41	jazakumullah	2
42	ussd	2
43	apps	2
44	bosster	2
45	app	2
46	smartfreen	2
47	message	2
48	boongan	2
49	google	2
50	add	2

Tabel A.3 Kata yang yang tidak dapat ditangani modifikasi penghapusan karakter berulang (lanjutan)

No	Kata	Frekuensi Muncul
51	penanganannya	2
52	good	2
53	followup	2
54	uufef	2
55	access	2
56	reboot	2
57	makassar	2
58	followback	2
59	speedtest	2
60	broo	2
61	xxl	2
62	handsfree	2
63	tweepscare	1
64	ttdftar	1
65	klikkrm	1
66	dibelaain	1
67	follback	1
68	account	1
69	masaaktif	1
70	blackberry	1
71	mengganguda	1
72	yummy	1
73	soo	1
74	mybankbiixltunai	1
75	boosternya	1
76	bii	1
77	terrulus	1
78	bantuannua	1
79	install	1
80	scobydoo	1
81	gantengganteng	1
82	hee	1
83	supermall	1
84	hii	1
85	https	1
86	approve	1
87	connected	1
88	hello	1

Tabel A.3 Kata yang yang tidak dapat ditangani modifikasi penghapusan karakter berulang (lanjutan)

No	Kata	Frekuensi Muncul
89	ketidaknyamannya	1
90	visaa	1
91	oalaah	1
92	difollow	1
93	verifikasiin	1
94	blablablaaujungnya	1
95	tanggapanitulah	1
96	disconnect	1
97	redeemnya	1
98	trafficnya	1
99	assalamualaikumuntuk	1
100	stts	1

Halaman ini sengaja dikosongkan

LAMPIRAN B

Berikut merupakan hasil klasifikasi yang didapatkan dengan menggunakan *tools* WEKA.

1. Pengujian tanpa penghapusan karakter berulang dengan menggunakan tahap penghapusan kata henti

```
Correctly Classified Instances        1696               70.6667 %
Incorrectly Classified Instances       704               29.3333 %
Kappa statistic                          0.56
Mean absolute error                      0.1956
Root mean squared error                  0.4422
Relative absolute error                 44      %
Root relative squared error             93.8083 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.725     0.134      0.73      0.725     0.727      0.795    1
                 0.613     0.219      0.583     0.613     0.598      0.697    0
                 0.783     0.087      0.818     0.783     0.8        0.848    -1
Weighted Avg.    0.707     0.147      0.71      0.707     0.708      0.78

=== Confusion Matrix ===

   a   b   c   <-- classified as
 580 198  22 |   a = 1
 193 490 117 |   b = 0
  22 152 626 |   c = -1
```

Gambar B.1 Pengujian tanpa penghapusan karakter berulang dengan menggunakan tahap penghapusan kata henti

2. Pengujian tanpa dengan metode illicker dengan menggunakan tahap penghapusan kata henti

```
Correctly Classified Instances        1710               71.25   %
Incorrectly Classified Instances       690               28.75   %
Kappa statistic                          0.5688
Mean absolute error                      0.1917
Root mean squared error                  0.4378
Relative absolute error                 43.125  %
Root relative squared error             92.8709 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.723     0.129      0.737     0.723     0.73       0.797    1
                 0.646     0.229      0.586     0.646     0.614      0.709    0
                 0.769     0.074      0.839     0.769     0.802      0.848    -1
Weighted Avg.    0.713     0.144      0.721     0.713     0.716      0.784

=== Confusion Matrix ===

   a   b   c   <-- classified as
 578 205  17 |   a = 1
 182 517 101 |   b = 0
  24 161 615 |   c = -1
```

Gambar B.2 Pengujian tanpa dengan metode illicker dengan menggunakan tahap penghapusan kata henti

3. Pengujian tanpa dengan metode jaccard dengan menggunakan tahap penghapusan kata henti

```
Correctly Classified Instances        1621               67.5417 %
Incorrectly Classified Instances       779               32.4583 %
Kappa statistic                          0.5131
Mean absolute error                      0.2164
Root mean squared error                  0.4652
Relative absolute error                 48.6875 %
Root relative squared error             98.6788 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.721     0.146      0.711     0.721     0.716      0.788    1
                 0.586     0.232      0.558     0.586     0.572      0.677    0
                 0.719     0.109      0.768     0.719     0.742      0.805    -1
Weighted Avg.    0.675     0.162      0.679     0.675     0.677      0.757

=== Confusion Matrix ===

   a   b   c   <-- classified as
 577 189  34 |   a = 1
 191 469 140 |   b = 0
  43 182 575 |   c = -1
```

Gambar B.3 Pengujian tanpa dengan metode jaccard dengan menggunakan tahap penghapusan kata henti

4. Pengujian modifikasi peghapusan karakter berulang dengan menggunakan tahap penghapusan kata henti

```
Correctly Classified Instances        1745               72.7083 %
Incorrectly Classified Instances       655               27.2917 %
Kappa statistic                          0.5906
Mean absolute error                      0.1819
Root mean squared error                  0.4265
Relative absolute error                 40.9375 %
Root relative squared error             90.4848 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.751     0.114      0.767     0.751     0.759      0.818    1
                 0.655     0.211      0.609     0.655     0.631      0.722    0
                 0.775     0.084      0.821     0.775     0.797      0.845    -1
Weighted Avg.    0.727     0.136      0.732     0.727     0.729      0.795

=== Confusion Matrix ===

   a   b   c   <-- classified as
 601 178  21 |   a = 1
 162 524 114 |   b = 0
  21 159 620 |   c = -1
```

Gambar B.4 Pengujian modifikasi peghapusan karakter berulang dengan menggunakan tahap penghapusan kata henti

5. Pengujian modifikasi peghapusan karakter berulang tanpa menggunakan tahap penghapusan kata henti

```
Correctly Classified Instances        1787               74.4583 %
Incorrectly Classified Instances       613               25.5417 %
Kappa statistic                          0.6169
Mean absolute error                      0.1703
Root mean squared error                  0.4126
Relative absolute error                 38.3125 %
Root relative squared error             87.5357 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.776     0.106      0.785     0.776     0.781      0.835    1
                 0.671     0.199      0.627     0.671     0.649      0.736    0
                 0.786     0.078      0.835     0.786     0.81       0.854    -1
Weighted Avg.    0.745     0.128      0.749     0.745     0.746      0.808

=== Confusion Matrix ===

   a   b   c   <-- classified as
 621 165  14 |   a = 1
 153 537 110 |   b = 0
  17 154 629 |   c = -1
```

Gambar B.5 Pengujian modifikasi peghapusan karakter berulang tanpa menggunakan tahap penghapusan kata henti

6. Pengujian tanpa praproses

```
Correctly Classified Instances        1559               64.9583 %
Incorrectly Classified Instances       841               35.0417 %
Kappa statistic                          0.4744
Mean absolute error                      0.2336
Root mean squared error                  0.4833
Relative absolute error                 52.5625 %
Root relative squared error            102.5305 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.67      0.15       0.691     0.67      0.68       0.76     1
                 0.51      0.236      0.519     0.51      0.515      0.637    0
                 0.769     0.139      0.734     0.769     0.751      0.815    -1
Weighted Avg.    0.65      0.175      0.648     0.65      0.649      0.737

=== Confusion Matrix ===

   a   b   c   <-- classified as
 536 224  40 |   a = 1
 209 408 183 |   b = 0
  31 154 615 |   c = -1
```

Gambar B.6 Pengujian tanpa praproses

7. Pengujian tanpa penghapusan karakter berulang

```
Correctly Classified Instances        1696               70.6667 %
Incorrectly Classified Instances       704               29.3333 %
Kappa statistic                          0.56
Mean absolute error                      0.1956
Root mean squared error                  0.4422
Relative absolute error                 44      %
Root relative squared error             93.8083 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.725     0.134      0.73      0.725     0.727      0.795    1
                 0.613     0.219      0.583     0.613     0.598      0.697    0
                 0.783     0.087      0.818     0.783     0.8        0.848    -1
Weighted Avg.    0.707     0.147      0.71      0.707     0.708      0.78

=== Confusion Matrix ===

   a   b   c   <-- classified as
 580 198  22 |   a = 1
 193 490 117 |   b = 0
  22 152 626 |   c = -1
```

Gambar B.7 Pengujian tanpa penghapusan karakter berulang

8. Pengujian tanpa stemming

```
Correctly Classified Instances        1682               70.0833 %
Incorrectly Classified Instances       718               29.9167 %
Kappa statistic                          0.5512
Mean absolute error                      0.1994
Root mean squared error                  0.4466
Relative absolute error                 44.875  %
Root relative squared error             94.7365 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.713     0.138      0.721     0.713     0.717      0.787    1
                 0.619     0.221      0.584     0.619     0.601      0.699    0
                 0.771     0.09       0.811     0.771     0.791      0.841    -1
Weighted Avg.    0.701     0.15       0.705     0.701     0.703      0.776

=== Confusion Matrix ===

   a   b   c   <-- classified as
 570 202  28 |   a = 1
 189 495 116 |   b = 0
  32 151 617 |   c = -1
```

Gambar B.8 Pengujian tanpa stemming

9. Pengujian tanpa konversi kata tidak baku

```
Correctly Classified Instances        1677               69.875  %
Incorrectly Classified Instances       723               30.125  %
Kappa statistic                          0.5481
Mean absolute error                      0.2008
Root mean squared error                  0.4481
Relative absolute error                 45.1875 %
Root relative squared error             95.0658 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.713     0.136      0.723     0.713     0.718      0.788    1
                 0.615     0.229      0.573     0.615     0.593      0.693    0
                 0.769     0.087      0.816     0.769     0.792      0.841    -1
Weighted Avg.    0.699     0.151      0.704     0.699     0.701      0.774

=== Confusion Matrix ===

   a   b   c   <-- classified as
 570 205  25 |   a = 1
 194 492 114 |   b = 0
  24 161 615 |   c = -1
```

Gambar B.9 Pengujian tanpa konversi kata tidak baku

10. Keseluruhan tahap praproses dengan metode llecker

```
Correctly Classified Instances        1721               71.7083 %
Incorrectly Classified Instances       679               28.2917 %
Kappa statistic                          0.5756
Mean absolute error                      0.1886
Root mean squared error                  0.4343
Relative absolute error                 42.4375 %
Root relative squared error             92.1276 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.746     0.123      0.753     0.746     0.75       0.812    1
                 0.638     0.218      0.594     0.638     0.615      0.71     0
                 0.768     0.084      0.821     0.768     0.793      0.842    -1
Weighted Avg.    0.717     0.141      0.722     0.717     0.719      0.788

=== Confusion Matrix ===

   a   b   c   <-- classified as
 597 182  21 |   a = 1
 177 510 113 |   b = 0
  19 167 614 |   c = -1
```

Gambar B.10 Pengujian keseluruhan tahap praproses dengan metode llecker

11. Keseluruhan tahap praproses dengan metode Jaccard

```
Correctly Classified Instances        1633               68.0417 %
Incorrectly Classified Instances       767               31.9583 %
Kappa statistic                          0.5206
Mean absolute error                      0.2131
Root mean squared error                  0.4616
Relative absolute error                 47.9375 %
Root relative squared error             97.9158 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.73      0.148      0.711     0.73      0.721      0.791    1
                 0.586     0.227      0.564     0.586     0.575      0.68     0
                 0.725     0.104      0.776     0.725     0.75       0.81     -1
Weighted Avg.    0.68      0.16       0.684     0.68      0.682      0.76

=== Confusion Matrix ===

   a   b   c   <-- classified as
 584 180  36 |   a = 1
 200 469 131 |   b = 0
  37 183 580 |   c = -1
```

Gambar B.11 Pengujian keseluruhan tahap praproses dengan metode Jaccard

12. Keseluruhan tahap praproses dengan modifikasi penghapusan karakter berulang

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        1787               74.4583 %
Incorrectly Classified Instances       613               25.5417 %
Kappa statistic                          0.6169
Mean absolute error                      0.1703
Root mean squared error                  0.4126
Relative absolute error                 38.3125 %
Root relative squared error             87.5357 %
Total Number of Instances             2400

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.776     0.106      0.785     0.776     0.781      0.835    1
                 0.671     0.199      0.627     0.671     0.649      0.736    0
                 0.786     0.078      0.835     0.786     0.81       0.854    -1
Weighted Avg.    0.745     0.128      0.749     0.745     0.746      0.808

=== Confusion Matrix ===

   a   b   c   <-- classified as
 621 165  14 |   a = 1
 153 537 110 |   b = 0
  17 154 629 |   c = -1
```

Gambar B.10 Pengujian keseluruhan tahap praproses dengan modifikasi penghapusan karakter berulang

LAMPIRAN C

Berikut merupakan hasil uji t paired yang didapatkan dengan menggunakan *tools* R yang dijabarkan pada Gambar C.1 dan C.2. d1 merupakan data yang diolah menggunakan penghapusan karakter berulang sedangkan d2 adalah data yang diolah menggunkanan modifikasi penghapusan karakter berulang. Pengujian dilakukan dengan perhitungan nilai p-value dengan convidence level 95%. Nilai p-value (sig) dibawah 0,05 menunjukkan adanya perbedaan signifikan antara kedua kelompok data. p-value = 0,00000000000000022 dan t = -16.640. Karena nilai p-value kurang dari 0,05 dan t bernilai positif, maka disimpulkan data pertama lebih rendah sigifikan nilainya dari data kedua.

```
> Ttest_data = read.csv (file.choose(), header=T)
> t.test (d1, data2, mu=0, alt="two.sided", paired=T, conf.level=0.95)
```

Gambar C.1 Alur Proses Uji t

```
        Paired t-test

data:  d1 and d2
t = -16.640, df = 13048, p-value = 0.00000000000000022
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.555155 -0.533051
sample estimates:
mean of the differences
            -0.544103
```

Gambar C.2 Hasil Uji t

Halaman ini sengaja dikosongkan

BIOGRAFI PENULIS

Fachrian Anugerah. Lahir di Surabaya, 28 November 1992. Merupakan anak kedua dari dua bersaudara. Penulis menempuh pendidikan formal mulai dari 1999-2005 di SD Muhammadyah 4 Surabaya. Pada tahun 2005-2008 di SMP Negeri 19 Surabaya, dan 2008-2011 di SMA Negeri 17 Surabaya. Tahun 2011 penulis melanjutkan jenjang pendidikan Strata 1 di jurusan Sistem Informasi, Fakultas Sains dan Teknologi, Universitas Airlangga Surabaya. Kemudian pada tahun 2015 penulis memutuskan untuk meneruskan pendidikan magister dan diterima sebagai mahasiswi di Institut Teknologi Sepuluh Nopember Surabaya pada jurusan Sistem Informasi yang berada dalam Fakultas Teknologi Informasi. E-mail: fachriananugerah.si@gmail.com