ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR –

ANALISIS SENTIMEN UNGGAHAN PADA TWITTER MENGGUNAKAN METODE GABUNGAN SUPPORT VECTOR MACHINE (SVM) DAN METODE K-MEANS DI PT.XYZ

SENTIMENT ANALYSIS OF POSTING ON TWITTER USING SUPPORT VECTOR MACHINE (SVM) AND K-MEANS IN PT.XYZ

HELENA ANGELITA DEPARI
NRP. 05211640000062

Dosen Pembimbing 1
Prof. Dr. Ir. Arif Djunaidy, M. Sc.

Dosen Pembimbing 2
Faizal Mahananto, S.Kom, M.Eng., Ph.D

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Elektro Dan Informatika CerdasInstitut Teknologi Sepuluh Nopember
Surabaya 2020

TUGAS AKHIR – IS184853

ANALISIS SENTIMEN UNGGAHAN PADA TWITTER MENGGUNAKAN METODE GABUNGAN SUPPORT VECTOR MACHINE (SVM) DAN METODE K-MEANS DI PT.XYZ

HELENA ANGELITA DEPARI
NRP. 052116400000062

Dosen Pembimbing 1
Prof. Dr. Ir. Arif Djunaidy, M. Sc.

Dosen Pembimbing 2
Faizal Mahananto, S.Kom, M.Eng., Ph.D

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Elektro Dan Informatika CerdasInstitut Teknologi Sepuluh Nopember
Surabaya 2020

Halaman ini sengaja dikosongkan

UNDERGRADUATE THESIS – IS184853

ANALISIS SENTIMEN UNGGAHAN PADA TWITTER MENGGUNAKAN METODE GABUNGAN SUPPORT VECTOR MACHINE (SVM) DAN K-MEANS DI PT.XYZ

HELENA ANGELITA DEPARI
NRP. 052116400000062

Dosen Pembimbing 1
Prof. Dr. Ir. Arif Djunaidy, M. Sc.

Dosen Pembimbing 2
Faizal Mahananto, S.Kom, M.Eng., Ph.D

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Elektro Dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 7 Juli 2020

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

ANALISIS SENTIMEN UNGGAHAN PADA TWITTER MENGGUNAKAN METODE GABUNGAN SUPPORT VECTOR MACHINE (SVM) DAN K-MEANS DI PT.XYZ

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer (S.Kom)
pada
Departemen Sistem Informasi
Fakultas Teknologi Elektro dan Informatika Cerdas (ELECTICS)
Institut Teknologi Sepuluh Nopember

Oleh

Helena Angelita Depari
05211640000062

Surabaya, 14 Agustus 2020

Kepala Departemen Sistem Informasi

Dr. Mudjahidin, ST., MT.
NIP. 197010102003121001

Halaman ini sengaja dikosongkan

LEMBAR PERSETUJUAN

ANALISIS SENTIMEN UNGGAHAN PADA TWITTER MENGGUNAKAN METODE GABUNGAN SUPPORT VECTOR MACHINE (SVM) DAN METODE K-MEANS DI PT.XYZ

TUGAS AKHIR

Disusun untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Elektro Dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember

Oleh :

HELENA ANGELITA DEPARI
NRP. 05211640000062

Disetujui Tim Penguji : Tanggal Ujian : 13 Juli 2020
Periode Wisuda : 26 September 2020

Prof. Dr. Ir. Arif Djunaidy, M. Sc.	**(Pembimbing I)**
Faizal Mahananto, S.Kom, M.Eng., Ph.D	**(Pembimbing II)**
Ahmad Mukhlason, S.Kom., M.Sc.,Ph.D	**(Penguji I)**
Raras Tyasnurita, S.Kom, MBA	**(Penguji II)**

ANALISIS SENTIMEN UNGGAHAN PADA TWITTER MENGGUNAKAN METODE GABUNGAN SUPPORT VECTOR MACHINE (SVM) DAN K-MEANS DI PT.XYZ

Nama : Helena Angelita Depari
NRP : 05211640000062
Departemen : Sistem Informasi ITS
Pembimbing I : Prof. Dr. Ir. Arif Djunaidy, M. Sc.
Pembimbing II : Faizal Mahananto, S.Kom, M.Eng., Ph.D

ABSTRAK

Kereta api merupakan salah satu transportasi umu yang dapat dijadikan pilihan, karena selain dapat mempersingkat waktu perjalanan, kereta api menyedikan berbagai jenis kelas denganharga yang terjangkau oleh masyarakat kebanyakan. Perusahaan penydia kereta api sudah berusaha untuk meningkatkan kualitas layannya, walaupun masih cukup banyak kritik yang diberikan terhadap pelayanan yang disediakan. Beberapa hal yang dikritisi ini dapat membuat masyarakat menjadi tidak loyal dan beralih untuk menggunakan jenis transportasi umum lainnya.

Biasanya, masyarakat menyampaikan pesan, saran, keluhan, dan kritik mereka di Twitter. Hal tersebut membuat Twitter menjadi salah satu media sosial yang dapat digunakan oleh perusahaan kereta api untuk bahan penilaian dan evaluasi terhdap layanan yang diberikan. Analisis sentimen merupakan metode klasifikasi yang dipilih untuk melakukan pengelompokan tweet sehingga dapat melihat tanggapan maupun gambaran masyarakat terkait pelayanan yang diberikan.

Tugas Akhir ini berkaitan dengan pembuatan dan implementasi analisis sentiment dengan menggunakan metode klasifikasi yang dipilih untuk melakukan pengelompokkan tweet sehingga dapat melihat tanggapan ataupun gambaran masyarakat terkait pelayanan yang dilakukan. Selain itu ada juga k-means yaitu cluster yang akan digunakan untuk digunakan untuk membantu pemberian label yang akan nantinya digunakan untuk proses analisa sentiment. Selain itu dicari kata- kata yang paling sering keluar, sehingga dapat melihat hal apa saja yang sering dikomentari atau ditanyakan oleh masyarakat. Hal ini dilakukan untuk meningkatkan kualitas relasi antar perusahaan dan pelangga yang nantinya dapat berpengaruh pada meningkatnya loyalitas dan juga kepuasan pelanggan. Implementasi pengelompokan diuji coba menggunakan k-means untuk mengelompokkan teks, dan analisis sentimen diuji coba menggunakan menggunakan SVM juga dilakukan. Hasil uji coba menunjukkan bahwa 96% hasil diperoleh untuk menguji data dan juga 93% untuk data pelatihan. Jadi dapat dikatakan bahwa SVM cukup baik dalam membuat model untuk data ini. Selain itu, terlihat juga bahwa publik paling sering bertanya di halaman Twitter. Adapun kritik, jumlahnya cukup minim, yaitu di bawah 20%, dan secara bertahap menurun.

Kata Kunci: Support Vector Machine, Kereta Api, Twitter, Sentiment Analysis.

ANALISIS SENTIMEN UNGGAHAN PADA TWITTER MENGGUNAKAN METODE GABUNGAN SUPPORT VECTOR MACHINE (SVM) DAN K-MEANS DI PT.XYZ

Nama : Helena Angelita Depari
NRP : 05211640000062
Departemen : Sistem Informasi ITS
Pembimbing I : Prof. Dr. Ir. Arif Djunaidy, M. Sc.
Pembimbing II : Faizal Mahananto, S.Kom, M.Eng., Ph.D

ABSTRACT

Train is one of the public transportation that can be used as an option, because besides being able to shorten travel time, trains provide various types of classes at prices that are affordable to the general public. Railway supply companies have made efforts to improve the quality of their services, although there are still quite a few criticisms given to the services provided. Several things that have been criticized can make people disloyal and switch to using other types of public transportation. Usually, people submit their messages, suggestions, complaints and criticisms on Twitter. This makes Twitter one of the social media that can be used by railway companies for assessment and evaluation of the services provided. Sentiment analysis is the classification method chosen for grouping tweets so that people can see responses and descriptions of the community regarding the services provided. This final project deals with the creation and implementation of sentiment analysis using the selected classification method for grouping tweets so that they can see the responses or images of the community regarding the services performed. In addition, there are also k-means, namely clusters that will be used to assist labeling which will be used for the sentiment analysis process. Apart from that, search for the words that come out most often, so that people can see what things are often commented on or asked about by the community. This is done to improve the quality of relationships

between companies and customers which in turn can have an effect on increasing loyalty and customer satisfaction. The implementation of clustering was tested using k-means to classify texts, and trial and error sentiment analysis using SVM was also carried out. The trial results showed that 96% of the results were obtained for testing the data and also 93% for the training data. So it can be said that SVM is quite good at modeling for this data. Apart from that, it was also seen that the public asked questions the most on the Twitter page. As for criticism, the number is quite minimal, at under 20%, and gradually decreasing.

Keywords : Train, Support Vector Machine, Clustering

SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan di bawah ini:

Nama	: Helena Angelita Depari
NRP	: 05211640000062
Tempat/Tanggal lahir	: Singkawang / 14 Juli 1998
Fakultas/Departemen	: FTEIC / Sistem Informasi
Nomor Telp/Hp/email	: 087889926377/helenaangelita1407@gmail.com

Dengan ini menyatakan dengan sesungguhnya bahwa penelitian/makalah/tugas akhir saya yang berjudul

ANALISIS SENTIMEN UNGGAHAN PADA TWITTER MENGGUNAKAN METODE GABUNGAN SUPPORT VECTOR MACHINE (SVM) DAN METODE K-MEANS DI PT.XYZ

Bebas Dari Plagiarisme Dan Bukan Hasil Karya Orang Lain.

Apabila dikemudian hari ditemukan seluruh atau sebagian penelitian/makalah/tugas akhir tersebut terdapat indikasi plagiarisme, maka saya bersedia menerima sanksi sesuai peraturan dan ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sesungguhnya dan untuk dipergunakan sebagaimana mestinya.

Surabaya, 13 Juli 2020

Helena Angelita Depari
NRP.05211640000062

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT, Tuhan Semesta Alam yang telah memberikan kekuatan sehingga penulis dapat menyelesaikan tugas akhir ini yang merupakan salah satu syarat kelulusan di Departemen Sistem Informasi Fakultas Teknologi Informasi dan Komunikasi Institut Teknologi Sepuluh Nopember Surabaya.

Terima kasih penulis pihak-pihak yang telah mendukung, memberikan saran, motivasi, semangat, dan bantuan baik berupa materiil ataupun moril demi tercapainya tujuan pembuatan tugas akhir ini. Secara khusus penulis akan menyampaikan ucapan terima kasih yang sedalam-dalamnya kepada :

1. Segenap keluarga besar terutama kedua orang tua adik penulis, Bapak Sander Sembiring, Ibu Butet Sitepu, dan adik Daniel Depari yang senantiasa mendoakan, memberikan motivasi dan semangat, sehingga penulis mampu menyelesaikan pendidikan sarjana ini dengan baik.
2. Oma penulis yaitu, Marlena Ginting yang senantiasa mendoakan, memberikan motivas, mendengarkan segala keluh kesah penulis selama di Surabaya dan Jakarta, sehingga penulis mampu menyelesaikan pendidikan sarjana ini dengan baik.
3. Ibu Mahendrawathi Er, S.T., M.Sc., Ph.D. selaku Kepala Departemen Sistem Informasi ITS, Bapak Nisfu Asrul Sani, S.Kom., M.Sc selaku Ketua Program Studi Sarjana Departemen Sistem Informasi ITS, serta seluruh dosen pengajar beserta staf dan karyawan Departemen Sistem Informasi ITS selama penulis menjalani perkuliahan.
4. Bapak Radityo Prasetianto Wibowo, S.Kom, M.Kom sebagai dosen wali penulis selama menempuh pendidikan di Departemen Sistem Informasi ITS yang selalu memberikan informasi dan kemudahan kepada penulis.

5. Bapak Prof. Dr. Ir. Arif Djunaidy, M. Sc. dan Bapak Faizal Mahananto, S.Kom, M.Eng., Ph.D selaku dosen pembimbing yang telah banyak meluangkan waktu untuk membimbing, mengarahkan, dan mendukung dengan memberikan ilmu, petunjuk, dan motivasi dalam penyelesaian tugas akhir ini.
6. Bapak Ahmad Mukhlason, S.Kom, M.Sc., Ph.D dan Ibu Retno Aulia Vinarti, S.Kom., M.Kom., Ph.D selaku dosen penguji yang telah memberikan kritik, masukan dan saran dalam pengerjaan tugas akhir ini.
7. Teman-teman laboratorium ADDI, RDIB, SE, dan MSI yang telah mempersilakan penulis bernaung dan mencari inspirasi dalam mengerjakan tugas akhir ini.
8. Rida, Santi, Tuti, dan Yunia yang telah mendengarkan curhatan, menyemangati, dan juga mendukung penulis dalam mengambil setiap pilihan yang diambil.
9. Seluruh pihak-pihak lainnya yang tidak dapat disebutkan satu per satu, yang telah membantu penulis selama perkuliahan hingga dapat menyelesaikan tugas akhir ini.

Penyusunan laporan tugas akhir ini masih jauh dari kata sempurna, sehingga penulis menerima adanya kritik maupun saran yang membangun untuk perbaikan di masa yang akan datang. Semoga buku tugas akhir ini dapat memberikan manfaat bagi pembaca.

Surabaya, 7 Juni 2020

Penulis

Helena Angelita Depari

DAFTAR ISI

DAFTAR GAMBAR

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Halaman ini sengaja dikosongkan

DAFTAR KODE

Halaman ini sengaja dikosongkan

BAB I
PENDAHULUAN

Pada Pada bab ini, akan diuraikan proses identifikasi masalah penelitian yang meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan penulisan beserta manfaatnya, serta relevansi pengerjaan tugas akhir.

1.1. Latar Belakang

Semakin meningkat jumlah populasi semakin meningkat pula kebutuhan akan transportasi. Transportasi inilah yang membuat manusia mudah dalam melakukan perpindahan satu tempat ketempat lainnya. Pada zaman sekarang ini, pilihan alat transportasi pun semakin banyak. Ragam alat transportasi ini juga memiliki kecepatan, dan tarif yang berbeda- beda. Selain itu, ragam transportasi ini juga dibedakan dari segi durasi waktu, dan tempat mereka berada.

Setiap orang yang ingin melakukan perjalanan jarak jauh antar negara atau antar kota dengan waktu yang relatif cepat, dapat memilih pesawat. Masyarakat yang ingin bepergiaan antar kota dengan waktu yang tidak terlalu cepat, tetapi harga yang ramah dikantong, dapat menggunakan kapal. Masyarakat yang ingin melakukan perjalanan antar kota yang membutuhkan waktu relative lebih lama, tetapi ingin terhindar dari kemacetan dapat menggunakan kereta api. Transportasi yang ada begitu beragam, sehingga setiap orang dapat memiliki alternatif transportasi yang cukup banyak.

Maka dari itu sebagai perusahaan kereta api mencoba meningkatka kualitas dan armada, sehingga masyarakat dapat menggunakan kereta api dengan nyaman dan dapat memilih kereta api sebagai transportasi yang hendak mereka gunakan untuk bepergian. Untuk mewujudkan hal itu, PT.XYZ memiliki layanan penyampaian keluhan baik dalam website perusahaan sendiri maupun melewati media sosial perusahaan yaitu Twitter. Twitter dipilih karena Twitter adalah salah satu media

sosial terbesar, dengan jutaan pengguna sehingga dapat memberikan wawasan yang luas. Twitter juga memiliki tingkat pertumbuhan pengguna bulanan yang cukup besar. Fakta- fakta tersebut, membuat minat para peneliti untuk memanfaatkan data komentar (*tweet*) dan melakukan teknik *mining* terhadap data tersebut. *Tweet* yang disampaikan melalui Twitter cukup banyak, sehingga hal itu memudahkan PT.XYZ dalam menganalisa kepuasan pelanggan mereka, agar PT.XYZ dapat memperbaiki dan meningkatkan layanan yang mereka miliki. Sebenarnya, PT.XYZ sudah melakukan perbaikan yang cukup signifikan, hanya saja salah kualitas layanan PT.XYZ harus terus meningkat kearah yang bisa meningkatkan kepuasan pelanggan. Tugas akhir ini, akan membantu untuk melakukan pengelompokan data tersebut. *Clustering* menggunakan *K-means* diharapkan dapat membantu mempermudah pemberian label. Selain melakukan pengelompokkan teks, dilakukan juga analisis sentimen untuk dapat menentukan unggahan bernada negatif dan positif dari setiap kelompokkan yang telah dihasilkan. Untuk melakukan analisis ini digunakan SVM agar dapat menghasilkan model untuk menganalisa *tweet* kedepannya. Algoritma SVM dianggap sebagai teknik optimasi yang dapat menentukan hubungan dalam satu set data, sesuai dengan pemecahan masalah optimasi. Sehingga SVM dapat memberikan solusi yang optimal [1].

Diharapkan dengan adanya tugas akhir ini, dapat membantu dalam menganalisa kepuasan pelanggan dan juga nantinya dapat membantu memberitahu apa saja pelayanan yang ditingkatkan atau pelayanan yang sering dikomentari. Hal tersebut dilakukan agar pelanggan tetap loyal terhadap PT.XYZ, dengan memilih kereta api sebagai transportasi yang diminati, walaupun banyak pilihan transportasi darat laiinnya. Dapat dilihat dinegara- negara maju, kereta tetap digemari oleh masyarakatnya. Hal itu akan dikaitkan dengan loyalitas pelanggan. Karena dengan perbaikan dengan berfokus pada

pelanggan, dipercaya dapat meningkatkan kualitas pelanggan dan juga menarik pelanggan lainnya yang belum pernah menggunakan transportasi ini menjadi pelanggan loyal karena layanan yang diberikan memuaskan, dan juga pihak perusahaan yang responsif dalam membantu pelanggan.

1.2. Rumusan Masalah

Berdasarkan uraian latar belakang yang telah dijabarkan, maka rumusan masalahnya adalah bagaimana membantu perusahaan untuk menilai seberapa tingkat kepuasan pelanggan secara cepat dan juga apa saja yang paling sering dikritik atau dikomentari pelanggan, sehingga loyalitas dan kepuasan pelanggan dapat ditingkatkan.

1.3. Batasan Masalah

Pada tugas akhir ini, terdapat beberapa batasan dikarenakan keterbatasan data dan waktu pengerjaan. Beberapa batasan itu adalah:

a. Data yang digunakan sebagai variabel dependen adalah data berupa teks yang diambil dari Twitter.
b. Data yang digunakan sebagai variabel independen merupakan *tweet* yang secara spesifik ditujukan kepada akun KAI.
c. Kasus yang diambil adalah *sentiment analysis*.
d. Pengunaan bantuan *k-means* hanya untuk mempermudah proses pembentukan kelas sentiment dengan cara memisahkan ke 3 *cluster*, lalu mendrop isi yang tidak sesuai pada cluster tersebut
e. Pengunaan bantuan *cluster* hanya untuk mempermudah proses pelabelan saja.

1.4. Tujuan

Berdasarkan latar belakang dan batasan permasalahan, maka tujuan tugas akhir ini adalah untuk membantu untuk menganalisa hal yang paling diinginkan pelanggan dan menganalisa pandangan serta penilaian pelanggan terhadap perusahaan. Selain itu tugas akhir ini bertujuan untuk

mengetahui apa saja hal yang paling sering dikomentari pelanggan sehingga loyalitas pelanggan dapat ditingkatkan. Tugas akhir ini dilakukan karena ingin memberikan informasi apa saja yang dominan dan bagaimana kepuasan pelanggan sehingga dapat mempengaruhi loyalitas pelanggan kereta api Indonesia.

1.5. Manfaat

Tugas akhir ini diharapkan dapat membawa manfaat untuk instansi sejenis ataupun instansi yang ingin menilai tingkat kepuasan pelanggan mereka. Manfaat yang didapatkan oleh pihak instansi adalah dapat terbantu untuk menganalisa kepuasan pelanggan dengan waktu yang relatif cepat menggunakan *tweet* yang masuk. Diharapkan dengan ini, pekerjaan pihak instansi dalam mengetahui apa saja hal- hal yang paling diperhatikan pelanggan dan layanan yang paling bermasalah bagi pelanggan.

1.6. Relevansi

Penelitian dari tugas ahir ini memliki relevansi dengan : Penelitian ini merupakan tema penelitian tugas akhir pada laboratorium Rekayasa Data dan Intelegensi Bisnis (RDIB) pada bidang *text mining*. Beberapa mata kuliah yang berkaitan dengan tugas akhir ini yang adalah Statistika, Analitika Bisnis, dan Penggalian Data. Penelitian ini memiliki relevansi dengan penelitian terdahulu yang meneliti terkait topik klasifikasi data dan *text mining* yang datanya bersumber dari Twitter.

Tugas akhir ini merupakan salah satu syarat kelulusan dari tahap sarjana di Departemen Sistem Informasi ITS.

Halaman ini sengaja dikosongkan

BAB II
TINJAUAN PUSTAKA

Bab ini menjelaskan tentang studi sebelumnya, yang mana penelitian sebelumnya yang terkait dengan tugas akhir ini. Tidak hanya itu, penjelasan dasar teori yang berisi tentang gambaran umum dari studi kasus dan teori mengenai metode yang akan digunakan dalam pengerjaan studi kasus tugas akhir ini.

2.1. Peneliatian Sebelumnya

Terdapat beberapa penelitian mengenai sentiment analysis menggunakan metode support vector machine (SVM). Oleh karena itu, dilampirkan beberapa penelitian sebelumnya untuk membantu pengerjaan tugas akhir dan juga untuk menunjang pengerjaan tugas akhir. Landasan teori akan memberikan gambaran umum dari acuan penjabaran tugas akhir ini.

Pada penelitian yang berjudul Sentiment Analysis in Twitter using Centroids, Clusters, and Sentiment Lexicons oleh Abeed Sarkerand and Graciela Gonzalez tahun 2016, peneliti melakukan penelitian terkait pembelajaran dengan pendekatan, menggunakan Support Vector Machines (SVMs) untuk klasifikasi sentimen pada Twitter. Penelitian ini fokusnya adalah untuk secara otomatis mengklasifikasikan polaritas kiriman Twitter dari tiga kategori yang telah ditentukan sebelumnya yaitu, positif, negatif dan netral. Dalam pendekatannya, peneliti menerapkan set fitur leksikal, semantik, dan distribusi yang telah diekstraksi. Penelitian ini memberikan masukan dalam pemakaian metode SVM dan dapat membantu pengerjaan tugas akhir [2].

Pada penelitian yang berjudul Sentiment Analysis for Twitter: TASS 2015 oleh Oscar S. Siordia dan Mario Graf tahun 2015, peneliti melakukan penelitian eksperimen klasifikasi polaritas Task of Spanish Tweets (TASS) untuk TASS 2015. Representasi tweet difokuskan pada fitur linguistik dan polaritas, filter kata-kata konten, aturan negasi,

dan lainnya. Selain itu, transformasi yang berbeda digunakan (LDA, LSI, dan TF-IDF) dan dikombinasikan dengan classifier SVM. Hasil menunjukkan bahwa representasi LSI dan TF-IDF meningkatkan kinerja klasifikasi SVM yang diterapkan. Penelitian ini menjadi salah satu alasan perlu dilakukan suatu kombinasi untuk meningkatkan kinerja SVM untuk analisis sentimen [3].

Pada penelitian yang berjudul Combining Classification and Clustering for Tweet Sentiment Analysis oleh Luiz F. S. Coletta, N´adia F. F. da Silva, Eduardo R. Hruschka Estevam R. Hruschka Jr tahun 2014, peniliti melakukan penelitian dengan tujuan agar dapat mengusulkan metode *cluster* \untuk membantu untuk mempermudah kinerja SVM agar memperoleh hasil yang memuaskan [4].

2.2. Dasar Teori

Sub bab ini berisi teori-teori yang mendukung serta berkaitan dengan tugas akhir yang disusun.

2.2.1. Tweet Pada Twitter

Twitter adalah media sosial yang banyak memuat tentang apa saja yang sedang diperbincangkan saat ini dan apa saja yang sedang marak terjadi saat ini [5]. Selain itu, Twitter juga merupakan salah satu layanan jejaring sosial yang memungkinkan penggunanya untuk mengirim dan membaca pesan dengan format teks yang memiliki batasan 140 karakter. Para pengguna Twitter, sering kali menuliskan keluhan ataupun apa yang mereka rasakan di laman pribadi mereka. Terkadang mereka menggunakan tanda pagar atau me-mention instansi atau seseorang yang ingin mereka kenakan pesan tersebut. PT.KAI merupakan salah satu instansi yang sering menerima keluhan ataupun saran dari pelanggannya melalui Twitter. Maka dari itu, dilakukan penelitian pada data Twitter dengan mengelompokkan tweet menggunakan data dari PT.KAI

2.2.2. *Data Mining*

Data Mining adalah kumpulan teknik otomatis yang lebih efisien dari pola yang sebelumnya tidak diketahui, tidak valid, baru, tetapi bermanfaat dan dapat dipahami dalam basis data besar. Pola tersebut harus diberi aksi agar dapat digunakan dalam pengambilan keputusan di perusahaan. Dalam data mining yang penting adalah [6]:

a. Data *Mining* adalah proses penemuan otomatisasi dalam volume data yang bervolume besar dan belum pernah diketahui polanya sebelumnya.
b. Data yang bervolume besar ini biasanya merupakan data historis dari suatu organisasi (*data warehouse*).
c. *Data Mining* biasanya dipakai untuk melakukan otomasi data yang bervolume besar.

2.2.3. K-Means

K-mean adalah algoritma pengelompokan yang sering sekali digunakan dalam pengelompokan teks. MacQueen James dan beberapa rekannya merupakan orang yang memperkenalkan algoritma ini pada tahun 1967. Pada awalnya *K-mean* menetapkan titik *centroid cluster* secara acak. *K-mean* sendiri akan mempartisi sekumpulan dokumen teks D = (d1, d2, d3, ..., dn) menjadi subset *cluster* K [6]. Algoritma ini menggunakan kesamaan maksimum untuk memasukkan setiap dokumen kedalam *cluster- cluster centroid* yang telah tersedia menggunakan suatu perhitungan. *K-mean text clustering* menggunakan jumlah klaster K dan pusat klaster awal untuk mengidentifikasi dokumen terkait yang ada di setiap kelompok atau klaster menggunakan persamaan kemiripan. Dimana rumus dari K-means sendiri adalah:

$$J_k = \sum_{k=1}^{K} \sum_{i \in C_k} (x_i - m_k)^2 \qquad (2.1)$$

Dengan $(x_1, x_2, \dots, x_n) = X$ merupakan data observasi $m_k = \sum_{i \in C_k} \frac{x_i}{n_k}$ merupakan pusat cluster C_k dengan n_k adalah jumlah

observasi dalam C_k. Untuk mempermudah langkah pengerjaan, maka berikut ini adalah pseudocode untuk K-means.

Algorithm 3 K-mean clustering algorithm

1: **Input**: A collection of text documents , K is the number of all clusters.
2: **Output**: Assign D to K.
3: **Termination criteria**
4: Randomly choosing K document as clusters centroid $C = (c_1, c_2, \ldots, c_K)$.
5: Initialize matrix X as zeros
6: **for** all d in D **do**
7: let $j = argmax_{k \epsilon \{1 to K\}}$, using the cosine similarity.
8: Assign d_i to the cluster j, A[i][j] = 1.
9:
10: Update the clusters centroids using Eq.
11: **end for**

Kode 2.1. Pseudocode K-means

K-means sendiri akan dipakai pada pengerjaan untuk membantu membentuk cluster dengan inputnya adalah kata- kata pada tweet yang ada. Setelah itu k-means akan melakukan pengelompokkan kata yang akan mengahasilkan kelompok kata-kata yang akan dianggap cluster kata postitif, cluster negatif, dan netral. Hal ini dilakukan karena tidak ada daftar kata-kat positif, negatif, dan netral dari pihak perusahaan. Bila cluster tidak memuat kata- kata yang diinginkan maka akan disesuaikan secara manual.

2.2.4. *Support Vector Machine*

SVM adalah salah satu dari sekelompok algoritma analisis pola yang dimaksudkan untuk menyelesaikan masalah klasifikasi nonlinier, regresi atau deteksi kebaruan [7]. Lalu SVM memiliki kemampuan untuk menyelesaikan masalah nonlinier. Algoritma SVM dianggap sebagai teknik optimasi yang dapat menentukan hubungan dalam satu set data, sesuai dengan pemecahan masalah optimasi. Sehingga SVM dapat memberikan solusi yang optimal [1]. SVM akan menemukan *hyperplane* terbaik yang memisahkan kelas-kelas pada input space [8]. Linear SVM bekerja dengan baik pada dataset yang dapat dengan mudah dipisahkan oleh hyperplane menjadi dua bagian. Tetapi, terkadang linear SVM sulit untuk

mengklasifikasi dataset yang kompleks. Ketika data kompleks, maka non-linear SVM yang biasa digunakan. Non-linear SVM adalah salah satu dari pola algoritma klasifikasi yang paling baik, karena dapat melakukan generalisasi maksimal ketika ingin memprediksi klasifikasi data yang sebelumnya tidak terlihat dibandingkan dengan data lain yang terlihat [9]. Model SVM mewakili data sebagai titik dalam ruang dibagi dengan garis / hyperplane. Fungsi pencarian *hyperplane* optimal ditunjukkan pada rumus (2.2) dan pertidaksamaan (2.3) [10]:

$$\frac{1}{2}\, wTw \;+\; C \sum \xi\, i\, i \tag{2.2}$$

$$\{(xi, yi)\}, yi(wTxi \;+\; b) \;\geq\; 1 \;-\; \xi i \tag{2.3}$$

w merupakan *weight vector* semenatara C merupakan *loss function* lalu ξi merupakan *slack variable/misclassification vector* i. xi merupakan *train vector* i ataupun lebih dikenal sebagai data *training*, lalu yi adalah kelas dari *train vector* i dan yang terakhir b merupakan nilai bias.

Pola xi akan bernilai negatif, jika masuk dalam pertidaksamaan:

$$\vec{w}.\vec{x} + b \;\leq\; -1 \tag{2.4}$$

Pola xi akan bernilai positif, jika masuk dalam pertidaksamaan:

$$\vec{w}.\vec{x} + b \;\geq\; 1 \tag{2.5}$$

Dalam melakukan SVM, diperlukan data sebagai input awalnya. Maka dipilihlah data vektor, karena dengan mentransformasi data text menjadi data vektor, maka akan mempercepat pengerjaan. SVM nantinnya akan diperlukan untuk membantu melakukan pelabelan pada data selanjutnya yang akan masuk. Karena SVM memerlukan data berlabel,

maka dilakukan dulu pelabelan, setelah itu barulah svm dapat mempelajari pola data.

2.2.5. *Vektor*

Vektor merupakan salah satu tipe data untuk input yang dapat dimasukkan kedalam SVM. Vektor sendiri dapat dihitung dengan cara manual ataupun computer. Vektor dapat dihitung dengan menggunakan TfidfVectorizer dan CountVector. Pada kali ini, perhitungan dilakukan dengan menggunakaan TfidfVectorizer. Vektor sendiri membutuhkan nilai dan arah, sementara pembobotan menggunakan TfidfVectorizer menghasilkan bobot pada teks. Sementara untuk arah digunakanlah cos 0, sehingga arah tersebut tidak akan merubah angka. Pembobotan dilakukan dengan menggunakan metode TF-IDF. TF-IDF memiliki formula sebagai berikut.

$$TF - IDF = TF \times IDF \qquad (2.6)$$

Rumus tersebut dapat dijabarkan menjadi term frequency dari fitur i pada dokumen j dikalikan dengan IDF. IDF sendiri merupakan kepanjangan dari Inverse Document Frequency. TF sendiri merupaka jumlah kemunculan term pada satu dokumen. IDF merupakan logJumlah seluruh dokumen. Semakin sering sebuah fitur muncul dalam sebuah teks, maka semakin besar pula bobot yang akan didapat, yang artinya maka akan semakin penting pula fitur tersebut

2.2.6. *Kernel Pada Support Vector Machine*

Jarang dijumpai kasus yang bersifat *linier separable*. Sehingga, SVM membutuhkan sebuah fungsi yang mampu membuat pemisah pada data yang tidak *linier*. Fungsi yang sering digunakan untuk mengatasi hal tersebut adalah fungsi kernel.

Pada kasus *nonlinier* SVM, data terlebih dahulu dipetakan oleh fungsi $\Phi(\vec{x})$ ke dalam ruang vektor yang berdimensi lebih tinggi. Setelah mendapat ruang vektor yang baru, kemudian *hyperplane* bisa dikonstruksikan untuk memisahkan kedua kelas sentimen yaitu positif maupun sentimen negatif didalam

tweet yang sudah ditentukan secara manual. *Hyperplane* bisa dikonstruksikan untuk memisahkan kedua kelas sentimen yaitu positif maupun sentimen negatif didalam *tweet* yang sudah ditentukan secara manual. Dengan pemakaian kernel, fungsi $\Phi(\vec{x})$ tidak perlu dicari. Hanya perlu menentukan *kernel* apa yang dipakai. Pada kasus ini *kernel* yang dipakai adalah *kernel*, RBF dengan rumus:

$$k(\overrightarrow{x_i}, \overrightarrow{x_j}) = exp\left(-\frac{\|\overrightarrow{x_i} - \overrightarrow{x_j}\|^2}{2\sigma^2}\right) \quad (2.7)$$

Untuk kernel ini akan ditambah *parameter* C dan γ.

Polynomial dengan rumus:

$$(X^T X + 1)^P \quad (2.8)$$

Untuk kernel ini akan ditambah parameter P.

Linear dengan rumus:

$$X^T X \quad (2.9)$$

Untuk kernel ini tidak ada yang harus ditambah. Fungsi kernel sendiri adalah untuk mempermudah pengerjaan. Dengan adanya kernel, tidak perlu diketahui secara detil wujud transformasi ke ruang dimensi yang lebih tinggi, cara mapping satu per satu suatu titik untuk berada pada suatu titik pada dimensi yang lebih tinggi. Sebenarnya bila melakuka SVM tanpa kernel, data akan diubah ke dimensi ruang yang jauh lebih tinggi daripada dimensi aslinya, sehingga membutuhkan waktu yang cukup lama dan proses yang cukup rumit.Tetapi dengan adanya kernel semua itu cukup dilakukan dengan menggunakan fungsi kernel.

2.2.7. *Text Preprocessing*

Setiap teknik analisa teks seperti, pencarian teks, pengelompokan teks, pemilihan fitur teks, dan lain-lain, perlu melakukan pengubahan terhadap isi dokumennya. Hal ini dilakukan agar dokumen tersebut dapat dikelola dengan algoritma yang mendasarinya. *Text Preprocessing* memainkan

peran yang sangat penting dalam teknik dan aplikasi *text mining*. Ini adalah langkah pertama dalam proses *text mining* [11]. Langkah- langkah untuk melakukan preprosesing biasanya terdiri dari *tokenization, filtering, lemmatization* dan *stemming*. Langkah- langkah tersebut akan dijelaskan jelaskan sebagai berikut:

a. Case folding

Pada langkah ini, akan dilakukan proses mengubah semua huruf dalam dokumen menjadi huruf kecil dan menghapus semua karakter yang bukan alfabet [12].

b. Tokenization

Tokenization adalah pemecahan kata menjadi potongan-potongan (kata / frasa) yang disebut token. Selain itu *tokenization* akan membuang karakter tertentu seperti tanda baca, sehingga kata- kata tersebut dapat berdiri sendiri.

c. Filtering

Filtering yang biasanya dilakukan pada dokumen adalah untuk menghapus beberapa kata yang tidak relevan. Selain itu *filtration* dilakukan untuk menghapus kata- kata yang tidak memberikan informasi sama sekali dan juga tidak memiliki relevasi, tetapi sering muncul dalam teks.

d. Lemmatization

Lemmatization adalah tahapan mempertimbangkan analisis morfologis kata-kata, yaitu mengelompokkan berbagai bentuk kata sehingga dapat dianalisa. Dengan kata lain, metode *lemmatization* mencoba memetakan setiap kata. Untuk *lemmatize* dokumen pertama kita harus menentukan aturan dari setiap kata dalam suatu dokumen.

e. Stemming

Metode *Stemming* bertujuan untuk mendapatkan akar dari suatu kata yang merupakan kata-kata turunan [13].

f. Pembobotan Text

Pembobotan text dilakukan setelah seluruh langkah diatas selesai. Langkah ini dilakukan dengan cara menggunakan *list*

kata positif dan list kata negatif. Lalu untuk setiap kata positif diberi nilai +1 dan setiap nilai negatif diberi nilai -1.

2.2.8. *Evaluasi Performa*

Evaluasi komparatif dilakukan dengan menggunakan satu ukuran evaluasi internal (pengukuran terkait kesamaan) dan empat ukuran evaluasi eksternal (akurasi (Ac), presisi (P), recall (R), dan ukuran-F (F)). Ukuran- ukuran ini adalah kriteria evaluasi umum yang digunakan pada domain *text clustering* untuk melakukan evaluasi akurasi

F-measure (F) adalah pengukuran umum yang digunakan dalam domain pengelompokan teks [14] . *F-measure* tergantung pada dua pengukuran: *precision* (2.10) dan *recall* (2.11). Precision dan recall adalah pengukuran umum yang digunakan di bidang text mining.

$$P(i,j) = \frac{ni,j}{nj} \tag{2.10}$$

$$R(i,j) = \frac{ni,j}{nj} \tag{2.11}$$

Dimana, (ni, j) pada rumus (2.10) dan (2.11) adalah jumlah anggota kelas i di *cluster* j, (nj) adalah jumlah anggota *cluster* j, dan ni adalah jumlah anggota kelas i. *F-measure* dihitung berdasarkan rumus (2.12):

$$F(j) = \frac{2\times P(i,j)\times R(i,j)}{P(i,j)\times R(i,j)} \tag{2.12}$$

Dimana, P (i, j) adalah anggota *precious* dari kelas i dalam *cluster* j, R (i, j) adalah anggota *recall* dari kelas i di *cluster* j, dan *F-measure* untuk semua *cluster* dihitung dengan mengikuti persamaan berikut [15]:

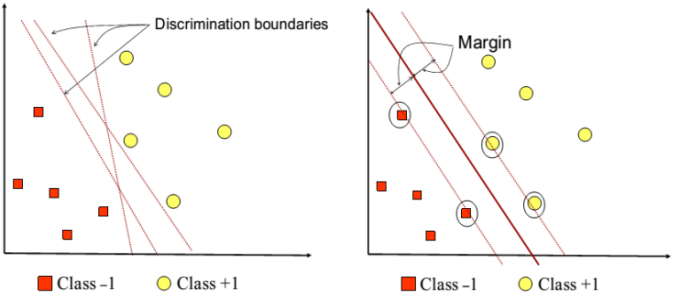
$$F = \sum_j \frac{nj}{n} max\ i\ \{n(i,j)\} \tag{2.13}$$

Pengukuran akurasi (AC) adalah salah satu pengukuran eksternal yang umum digunakan untuk menghitung persentase dokumen yang sudah disetujui kebenarannya untuk setiap *cluster* sesuai dengan persamaan (2.14) [16]:

$$AC = \frac{1}{n}\sum_{i=1}^{k} ni, i \quad (2.14)$$

2.2.9. *Analisa Sentimen*

Klasifikasi sentimen berguna untuk menentukan apakah suatu kalimat atau teks memiliki opini positif atau negatif. Analisa sentimen ini berguna untuk mengetahui tentang "Sudah seberapa puaskah pelanggan?" dan juga pandangan pelanggan terhadap kinerja perusahaan. Informasi penting ini dapat digunakan untuk membantu ataupun mempengaruhi setiap proses pengambilan keputusan (untuk bagian apa pun) dalam pemerintah maupun perusahan [3]. Pada analisis sentimen ini digunakan SVM sehingga nantinya akan membentuk dua kelas seperti yang ada pada gambar 2.2:



Gambar 2.1. SVM

2.2.10. *Customer Relationship Management*

Customer relationship management (CRM) adalah kombinasi dari manusia, proses, dan teknologi yang Bersatu untuk memahami keinginan pelanggan perusahaan. CRM adalah pendekatan terpadu untuk mengelola hubungan dengan berfokus pada retensi pelanggan dan pengembangan hubungan [17]. CRM telah berkembang dari sisi kemajuan teknologi informasi maupun dari sisi perubahan- perubahan yang dilakukan oleh organisasi dalam prosesnya yang berpusat pada pelanggan. Perusahaan yang berhasil menerapkan CRM akan mendapat banyak keuntungan baik dalam loyalitas pelanggan

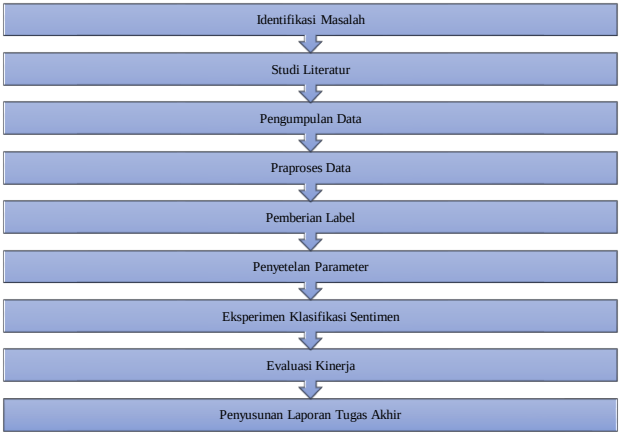
dan juga peningkatan profitabilitas jangka Panjang. Hanya saja banyak perusahaan yang belum memikirkan dari sisi peanggan. Fokus utama perusahaan kebanyakan hanya terkait keuntungan secara keuangan saja. CRM sendiri memiliki banyak tipe salah satunya adalah *social relationship management*.

2.2.11. *Social Customer Relationship Management*

Social customer relationship management atau CRM 2.0 adalah bicara terkait manusia dan hubungan- hubungannya serta tuntutan untuk tetap berfokus pada pelanggan. Media sosial memberikan dampak yang sangat signifikan terhadap keberhasilan dari penerapan SCRM sendiri. Media sosial menjadi elemen penting dalam aktivitas CRM perusahaan karena dengan adanya media sosial, perusahaan dapat lebih memahami pelanggannya yang kemudian berdampak pada meningkatnya loyalitas pelanggan dan menghasilkan profit yang lebih tinggi. SCRM sendiri mengusulkan model konseptual untuk mengatasi hubungan antara manajemen hubungan pelanggan, teknologi media sosial, keterlibatan pelanggan, kata-kata positif dari mulut ke mulut dan loyalitas pelanggan.

BAB III
METODOLOGI PENGERJAAN

Dalam bab ini, akan dijelaskan tahapan- tahapan yang akan dilalui dalam pembuatan tugas akhir ini. Tahapan- tahapan ini disesuaikan dengan tahapan- tahapan pengerjaan tugas akhir ini mulai dari mengidentifikasi masalah, pemrosesan data, dan seterusnya. Adapun tahapan- tahapan tersebut, akan dituangkan di dalam metodologi dan akan dijelaskan tiap tahapannya di bab ini.



Gambar 3.1 Metodologi

3.1. Identifikasi Masalah

Dalam tahap ini dilakukan identifikasi terkait permasalahan-permasalahan yang akan diselesaikan pada tugas akhir ini. Permasalahan itu terkait klasifikasi dan pencarian homogenitas pada tweet- tweet yang ditujukan kepada KAI. Masalah yang ditemui secara teknis adalah belum adanya pelabelan dan dibutuhkan orang dalam perusahaan atau orang yang cukup kompeten untuk dapat mengklasifikasikan tweet- tweet tersebut. Data yang digunakan adalah data lebih dari tiga tahun, mulai tahun 2015.

3.2. Studi Literatur

Setelah pengidentifikasian masalah, maka dilanjutkan dengan studi literatur. Pada langkah ini, penulis akan mencari berbagai jurnal, buku dan penelitian sebelumnya yang terkait. Langkah ini sangat diperlukan, karena akan memudahkan peneliti untuk memahami teori dari metode, optimasi, dan sifat data yang akan diklasifikasi pada tugas akhir ini. Hasil dari studi literature ini dapat dilihat di sub bab 2.2.

3.3. Pengumpulan Data

Setelah studi literatur, maka langkah selanjutnya adalah pengumpulan data dan informasi. Pada tahap ini, penulis akan mengumpulkan data-

data yang dibutuhkan untuk mengerjakan penelitian ini. Data tersebut berupa file excel dari proses *crawling* pada data Twitter. Informasi yang dibutuhkan baik untuk membantu pengolahan data dan pemecahan masalah pada tugas akhir ini akan dikumpulkan pada langkah ini. Data yang digunakan adalah data dari Twitter KAI pada tahun juli 2015 – Oktober 2019.

3.4. Praproses Data

Pada langkah ini, data yang telah didapat dari tahap sebelumnya yang masih merupakan data mentah akan diolah. Pengolahan data ini dilakukan untuk mengatasi masalah missing value, data yang terduplikasi, dan pembersihan tweet. Setelah permasalahan terkait selesai, data dapat dilanjutkan ke tahap normalisasi data. Selain itu harus ada normalisasi untuk menormalkan variabel agar dapat mengekspresikannya dalam rentang nilai yang sama. Dengan kata lain, normalisasi berarti menyesuaikan nilai yang diukur pada skala yang berbeda ke skala umum. Karena memang masing masing variabel memiliki rentang yang berbeda (max,min) sehingga perlu disetarakan atau normalisasi. Untuk langkah- langkah secara lebih terperincinya dapat dilihat pada subbab 2.2.8.

Missing Value dalam data yang didapat terdapat beberapa data yang ada seperti pada table dibawah. Untuk data- data seperti

ini akan dilakukan penghapusan, karena tidak memiliki teks yang dapat diklasifikasi ataupun diproses.

Tabel 3.1. Teks Kosong

Asal Kasus	**Deskripsi**
Social media	**#REF!**

Tabel 3.1 menunjukkan contoh baris yang kosong, ketika diklik makatidak keluar isinya. Data seperti itu terdapat cukup banyak, sehingga perlu dilakukan penghapusan data.

3.5. Pembersihan Tweet

Pada tahapan ini akan dilakukan pembersihan *tweet* dengan melakukan beberapa tahapan. Tahapan awalnya adalah menghapus *mention, link,* dan *hashtag* yang ada.

Tabel 3.2. Tweet Mentahan

isi
-KAI121 ke purwakarta dri sta kota ya? 5:04 PM - 14 Aug 2015 åá
-KAI121 gampang kok, dari web resmi nya aja langsung di http://www.kereta-api.co.id atau download aplikasi nya juga bisa di Hp..
-KAI121 Selamat pagi Kereta Api... Semoga selalu terbaik... petugas pelayanan tiket Garum Blitar tidak pernah ... http://tmi.me/1f8P33
P : kereta #tarifkhusus itu berlaku gak kalo naik dari stasiun Ciamis?

Setelah itu, akan dihapus pula angka- angka dan simbol-symbol yang ada pada setiap tweet yang ada. Pembersihan ini dilakukan agar mempermudah dalam melakukan pembuatan klaster, sehingga tidak banyak kata- kata atau simbol yang masuk. Hal ini akan mempermudah dalam memeriksa isi klaster yang terbentuk dan juga proses pelabelan.

Tabel 3.3. Tweet Bersih

isi
ke purwakarta dri sta kota ya PM Aug
gampang kok, dari web resmi nya aja langsung di atau download aplikasi nya juga bisa di
Selamat pagi Kereta Api... Semoga selalu terbaik... petugas pelayanan tiket Garum Blitar tidak pernah
P kereta itu berlaku gak kalo naik dari stasiun

Setelah melakukan tahapan tersebut maka akan didapat tweet yang bersih seperti tabel 3.3. Selain melakukan pembersihan data tweet, dilakukan juga perubahan huruf, semua huruf besar akan dijadikan huruf kecil, ahar lebih mudah untuk dikenali ketika memiliki arti yang sama.

Tabel 3.4. Tweet dengan Huruf yang Sama

isi
ke purwakarta dri sta kota ya pm aug
gampang kok, dari web resmi nya aja langsung di atau download aplikasi nya juga bisa di
selamat pagi kereta api... semoga selalu terbaik... petugas pelayanan tiket garum blitar tidak pernah
p kereta itu berlaku gak kalo naik dari stasiun

Setelah semua huruf dijadikan kedalam huruf kecil, maka kata-kata yang sama akan lebih mudah dikelompokkan dan diekstrak. Karena perbedaan huruf dapat mempengaruhi pengelompokkan kata.

3.6. Pemberian Label

Pemberian label dilakukan dengan mendata kata- kata dalam yang bernada positif dan negatif pada kumpulan *tweet* yang mengarah pada perusahaan, agar mudah dalam melakukan pemberian label positif, negatif, dan netral setiap tweet yang ada. Pendataan tersebut dilakukan dengan membentuk klaster kata- kata dari *tweet* yang ada, lalu akan ditentukan kluster positif dan negatifnya. Pembentukan klaster tersebut memerlukan suatu metode klaster, sehingga dipakailah *K*-

means untuk melakukan clustering tersebut. Setelah itu barulah dapat digunakan klaster tersebut untuk melakukan pelabelan. Selain itu ada data yang mengandung daftar kata- kata positif dan negatif yang sudah tersedia, sehingga tinggal diimplementasikan saja. Setelah itu tweet akan diberi nilai berdasarkan kata- kata yang terdapat didalamnya. Pada pengolahan data label negatif akan bernilai -1, positif bernilai 1 , dan netral bernilai 0 . Pelabelan itu akan diberikan seperti yang ada pada tabel dibawah ini:

Tabel 3.5. Tweet Hasil Pelabelan

Deskripsi	Sentimen	Nilai
tolong di cek salah satu stasiun di surabaya menjual tiket tanpa nomor terus penumpangnya nyolong tempat penumpang yang ada nomornya.	Negatif	-1
semalem naik ka taksaka malam gambir yogya sampe jogja lebih cepat dari jadwal tiba terima kasih sangat memuaskan	Positif	+1
kalo dari kutoarjo ke purwokerto prameks sampe kutoarjo kan min?	Netral	0

3.7. Penyetelan Parameter

Penyetelan parameter ini akan dilakukan untuk menyetel parameter C dan mengganti kernel yang digunakan . Penyetelan ini membutuhkan beberapa kali percobaan sehingga menemukan nilai terbaik untuk menghasilkan *hyperplane* terbaik pada SVM. Selain itu digunakan pula kernel RBF, linear, dan polynomial. Pada kernel itu aka nada dua nilai yang paling mempengaruhi yaitu C dan γ. γ mendefinisikan seberapa besar pengaruh setiap data terhadap hasil permodelan. Sementara C berbicara tentang seberapa halus permukaan yang dihasilkan serta seberapa tingkat kebenaran model

yang dihasilkan. Semakin besar nilai C maka akan semakin benar model yang dihasilkan, walau proses pengerjaan akan menjadi semakin lama. Pada kernel polynomial, nilai yang dijadikan penentu adalah derajat P.

3.8. Eksperimen Klasifikasi Sentimen

Pada tahap ini, akan dilakukan klasifikasi sentimen data yang dimiliki menggunakan metode SVM. Pengerjaan ini menggunakan bahasa pemrograman Python. SVM yang dipakai menggunakan *kernel* Radial Basis Function (RBF), Polinonial, serta linear. Untuk langkah- langkah lebih terperinci dapat dilihat pada subbab 2.2.2, 2.2.5, dan 2.2.6.

3.9. Evaluasi Kinerja

Setelah seluruh proses training dijalankan, maka akan dilakukan proses analisis terhadap model yang dihasilkan. Model yang dihasilkan akan dievaluasi performanya. Analisis ini termasuk dengan pengujian terhadap sekumpulan data uji. Proses ini dilakukan untuk mengukur sejauh mana model mampu klasifikasi teks dengan tepat. Untuk evaluasi ini langkah terperincinya dapat dilihat pada subbab 2.2.7

3.10. Penyusunan Tugas Akhir

Pada tahapan penyusunan tugas akhir ini akan dilakukan penulisan laporan tugas akhir dengan tujuan sebagai dokumentasi pelaksanaan penelitian tugas akhir ini. Laporan tersebut mencakup :

a. Bab I yaitu Pendahuluan yang berisi latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat pengerjaan tugas akhir ini.

b. Bab II yaitu Tinjauan Pustaka yang menjelaskan mengenai penelitian terdahulu dengan topik serupa dan juga teori-teori penunjang permasalahan yang dibahas pada tugas akhir ini.

c. Bab III yaitu Metodologi Penelitian yang akan menjelaskan tentang tahapan yang akan dilakukan dalam pengerjaan penelitian tugas akhir ini.

d. Bab IV yaitu Desain yang akan menjelaskan mengenai bagaimana rancangan dari penelitian tugas akhir yang terdiri dari mendefinisikan ruang lingkup penelitian dan bagaimana penelitian ini akan dilakukan.

e. Bab V yaitu Implementasi yang akan menjelaskan tentang proses pelaksanaan penelitian dan pembuatan model untuk klasifikasi.

f. Bab VI yaitu Hasil dan Pembahasan yang menjelaskan tentang pembahasan pengerjaan di tugas akhir ini.

g. Bab VII yaitu Kesimpulan dan Saran yang menjelaskan tentang kesimpulan dan saran untuk menyempurnakan tugas akhir ini.

BAB IV
DESAIN PERMODELAN

Pada bab ini akan menjelaskan mengenai bagaimana rancangan dari penelitain tugas akhir yang terdiri dari mendefiniskan ruang lingkup penelitian dan bagaimana penelitain ini akan dilakukan.

4.1. Perencanaan Persiapan Perangkat

Pada tahap ini akan dilakukan persiapan perangkat yang digunakan. Perangkat yang digunakan adalah laptop dan usb. Selain itu diperlukan beberapa *library*. Perangkat yang digunakan sebagai berikut.

Tabel 4.1. Persiapan Perangkat Keras

Perangkat Keras	Komponen	Spesifikasi
Laptop Asus Zenbook 13 UX333F	Prosesor	Intel® Core™ i5-8265U 1.6GHz (Turbo up to 3.9GHz)
	Ram	8 GB
	Sistem Operasi	Windows 10 (64bit)
USB Toshiba	Kapasitas	16 GB

4.2. Desain Crawler

Pada tahap ini akan dilakukan proses *crawling*, proses ini dilakukan untuk mencari tambahan data, agar data yang sebelumnya dapat lebih dicukupkan. Data awal yang dimiliki hanyalah data tahun 2016, sehingga data dinilai kurang relevan bila dibuat Analisa untuk membantu menyelesaikan permasalahan pada saat ini. Dataset yang akan diambil adalah *tweet* terkait KAI121. Proses ini diharapkan dapat membantu melengkapi data yang kurang yaitu data awal 2017 sampai data akhir 2019. *Crawling* ini akan dilakukan penggunakan kode perintah dengan bahasa pemrograman Python.

4.3. Perencanaan Pengumpulan Data

Pada tahapan ini dilakukan pengumpulan data untuk dipakai ke proses selanjutnya. Tetapi, terdapat kendala, karena banyaknya data masih kurang cukup dan belum sampai yang terbaru. Sehingga, perlu dilakukan *crawling* twitter KAI untuk mendapat data. Akhirnya, didapatlah data sampai Oktober 2019, sehingga dapat dikatakan, data sudah cukup banyak dan sudah sampai yang cukup terbaru, sehingga bila dilakukan analisa, hasilnya akan lebih relevan. Selain data *tweet*, ditengah pengerjaan akan dilakukan juga penginputan data daftar kata positif, negatif, dan netral yang didapat dari hasil *clustering* pada kata- kata yang terdapat pada kumpulan *tweet*.

4.4. Perencanaan Pemasukkan Library dan Data

Pada tahap ini akan dimasukkan keseluruhan data, sehingga dapat dilihat bahwa data memerlukan proses pembersihan, proses eksplorasi, proses normalisasi, dan proses lainnya. Maka dari itu, diperlukan import beberapa library untuk dapat melakukan proses tersebut. Import library dilakukan diawal, tetapi bila membutuhkan bantuan library lain disaat proses pengerjaan, dapat melakukan import juga. Selain library, data yang telah dikumpulkan masukkan juga.

4.5. Perencanaan Praproses Data

Pada tahap ini diketahui bahwa data belum siap diolah. Hal tersebut dikarenakan data tersebut belum dibersihkan, belum diberi nilai, dan juga belum berlabel. Perlu dilakukan penghapusan duplikat, pembersihan, dan penilaian.

4.5.1. Perencanaan Pemotongan Kolom

Pada tahap ini dapat dilihat tidak semua kolom dan informasi dibutuhkan dalam pengolahan data. Sehingga beberapa data yang tidak akan digunakan perlu dihapus. Hal tersebut dilakukan agar pengolahan dapat lebih fokus dan lebih mudah.

4.5.2. Perencanaan Penghilangan Duplikat

Pada tahap ini dapat dilihat bahwa data yang sudah didapat dikumpulkan, lalu dicari apakah ada data yang terduplikasi. Data yang terduplikasi akan dihapus sehingga akurasi bisa semakin tepat. Selain akurasi, penghapusan data duplikat juga

akan mengurangi lama pengolahan data. Dipercaya bila data berkurang, maka pengolahan pun butuh waktu yang lebih sedikit.

4.5.3. Perencanaan Penghapusan Data yang Hilang

Pada tahap ini, akan dilakukan pengolahan dataset, ada kemungkinan terdapat missing value. Missing value tersebut dapat berupa teks yang tidak tersedia, #NAME, #REF, dan ada pula baris yang kosong. Hal tersebut perlu dihilangkan agar tidak mempengaruhi jalannya proses pengolahan data dan juga mengurangi jumlah data, sehingga proses pengolahan lebih cepat.

4.5.4. Perencanaan Pembersihan Dataset

Padat tahap ini akan dilakukan pembersihan data dari duplikasi akan dibersihkan. Pembersihan ini dilakukan dengan menghapus tanda baca, *mention*, *link*, *hastag*, serta missing value yang tidak mempengaruhi hasil akhir. Karena pada dasarnya yang diperlukan hanyalah tweet yang bersih tanpa tanda baca, *mention*, *link*, dan *hashtag*. Selain hal tersebut, simbol- simbol dan angka dihapus juga. Hal tersebut dilakukan agar proses penglompokkan lebih mudah. Selain itu, data tersebut tidak dibutuhkan dan tidak mempengaruhi hasil Analisa. Selain itu seluruh huruf yang ada diubah menjadi huruf kecil semua. Hal ini dilakukan karena huruf kapital dan huruf kecil dapat mempengaruhi penilaian data, karena kata yang memiliki perbedaan besar kecil huruf, bisa dianggap dua kata yang berbeda.

4.6. Perencanaan Tokenisasi

Pada tahap ini, *tweet* yang ada akan diproses serta dipisah satu persatu. Kata- kata tersebut dimasukkan kedalam array. Pemisahan kata tersebut diharapkan akan mempermudah ketika melalui langkah pencocokan kata dengan daftar kata untuk mendapatkan nilai. Baik dicocokkan kekata- kata positif dan kata- kata negatif. Pemisahan ini juga diperlukan dalam mempermudah pembuatan itu sendiri

4.7. Perencanaan Vektorisasi

Pada tahap ini akan dilakukan vektorisasi. Vektorisasi sendiri merupakan tahapan yang dilakukan sebelum pengelompokkan. Bagian ini merupakan langkah dimana data teks yang ada ubah menjadi bentuk data vektor. Vektorisasi ini dilakukan untuk mempercepat pengerjaan pengolahan data. Hal itu didasarkan pada jumlah data yang begitu banyak lebih dari 100.000 yang bila tidak dibantu proses ini akan memakan waktu yang cukup lama. Hasil vektorisasi ini akan digunakan sebagai input untuk SVM sendiri.

4.8. Perencanaan Pembentukan Klaster Kata

Pada tahap ini akan dilakukan pengelompokan menggunakan bantuan metode k-means. Input untuk k-means sendiri adala data teks yang diubah menjadi data vektor. Setiap data yang ada dimasukkan kedalam klaster yang berbeda dengan metode k-means. Klaster yang sudah terbentuk nantinya akan diuji. Pengujian dilakukan untuk mengetahui seberapa dekat relasi antara objek dalam sebuah cluster dan seberapa jauh sebuah cluster terpisah dengan cluster lain. Metode pengujian yang digunakan adalah Silhouette Coefficient. Tujuan dari tahap ini adalah memudahkan untuk melakukan pelabelan. Langkahnya adalah, teks dimasukkan kedalah tiga klaster. Setelah itu, klaster ditentukan mana yang termasuk klaster positif, negatif, dan netral. Kata- kata yang sudah dimasukkan kedalam cluster akan diperiksa kembali agar memastikan bahwa tidak ada kata yang ditempatkan ditempat yang salah. Hal tersebut dilakukan agar tingkat kesalahan dapat berkurang. Kata- kata yang sudah dipisahkan itu ditambahkan dengan kata- kata dari domain bebas yang telah ditambahkan sebelumnya. Tahap ini dilakukan sebenarnya untuk mempermudah pelabelan. Data yang sangat banyak kurang memungkinkan untuk dilabeli satu persatu, sehingga digunakanlah bantuan pengelompokkan kata ini. Pengelompokkan kata ini akhirnya akan menjadikan tiga klaster, setiap klaster akan dilakukan mengecekkan Kembali apakah benar sudah sesuai dengan yang diinginkan, bila tidak,

akan ditambahkan dan dikurangi secara manual isi klaster tersebut. Agar pelabelan dapat dilakukan secara otomatis. Sehingga tinggal dilakukan pengecekan akhir saja.

4.9. Perencanaa Pengecekan Kata yang Sering Muncul

Pada tahap ini akan dilakukan tahapan melihat kata- kata yang paling sering muncul dengan cara menampilkan kata tesebut dan juga jumlah kemunculannya. Kata yang paling sering muncul ini akan dilakukan untuk melihat masalah apa saja yang paling sering dibahas atau hal yang paling sering dikomentari oleh pengguna Twitter pada laman perusahaan.

4.10. Perencanaan Penilaian Tweet

Pada tahap ini, tweet yang telah dimasukkan kedalam array akan diberi nilai. Tweet tersebut diberi nilai dengan cara dicocokkan kata perkata dengan kata- kata yang ada baik didaftar kata negatif dan juga didaftar kata positif. Daftar kata tersebut terdapat dari hasil kata- kata yang terdapat pada Twitter perusahaan terlebih dahulu. Penilaian didapat dengan mencocokkan kata setiap tweet kedalam daftar kata positif dan daftar kata negatif. Bila kata tersebut ada didaftar kata negatif akan diberi nilai -1, tetapi jika kata tersebut ada pada daftar kata positif ketika dilakukan pencocokan maka akan diberi nilai +1, jika tidak ada dikeduanya maka akan diberi nilai 0. Setelah dinilai, maka akan dijumlahkan nilai setiap kata yang ada pada satu tweet, sehingga setiap tweet yang ada akan ada total nilainya. Total nilai tersebut yang nantinya akan menunjukkan positif, negatif, dan netral. Semisal ada tweet “kereta datang tepat waktu”. Maka akan dicek satu persatu katanya, kata pertama adalah ‘kereta’, kata tersebut tidak ditemukan dalam daftar kata negatif, maupun daftar kata positif, sehingga bernilai 0. Kata kedua adalah ‘datang’, kata tersebut tidak ditemukan dalam daftar kata negatif, maupun daftar kata positif, sehingga bernilai 0. Kata ketiga adalah ‘tepat’, tepat merupakan kata yang terdapat pada daftar kata positif, sehingga diberi nilai +1. Lanjut lagi kata terakhir yaitu ‘waktu’, kata tersebut tidak ditemukan

dalam daftar kata negatif, maupun daftar kata positif, sehingga bernilai 0. Sehingga ketika ditotal maka nilai untuk tweet tersebut adalah 0 + 0 + 1 + 0 =1. Sehingga tweet terebut bernilai +1.

4.11. Perencanaan Pemberian Label

Pada tahap ini akan dilakukan pemberian label. Hal tersebut dilakukan untuk membantu Analisa sentiment. SVM sendiri merupakan metode klasifikasi yang mebutuhkan label, sehingga *tweet* yang ada diberi label terlebih dahulu. Untuk itu, perlu dilakukan pemberian label pada setiap tweet yang ada. Pemberian label itu dilakukan dengan memberikan label positif terhadap kelompok dengan nilai tweet yang lebih dari sama dengan +1, memberikan label negatif terhadap satu kelompok dengan nilai kurang dari sama dengan -1, serta memberikan label netral terhadap satu kelompok dengan nilai 0.

Tabel 4.2. Penjelasan Sentimen

Sentimen	Penjelasan
Negatif	Sentimen negatif merupakan kalimat yang mengandung makna negatif. Biasanya berupa komentar ataupun keluhan.
Netral	Sentimen netral merupakan kalimat yang mengandung makna netral yang tidak memihak kearah negatif maupun positif sedikitpun. Biasanya merupakan kata tanya atau hanya pemberitahuan.
Positif	Sentimen positif merupakan kalimat yang mengandung makna positif. Biasanya merupakan komentar pujian atau bisa juga merupakan ungkapan.

4.12. Perencanaan Penyetalan Parameter

Support Vector Machine sendiri memiliki parameter tertentu. Parameter ini perlu dilakukan penyetelan agar dapat mendapatkan hasil analisa yang baik. Pada SVM terdapat dua

parameter yang dapat disetel untuk meningkatkan akurasi , kedua parameter tersebut adalah nilai C. Sebelum penyetelan parameter, diperlukan kernel untuk melakukan SVM ini. Secara otomatis, *kernel* RBF, linear, dan polinonial yang sudah terpasang. Kernel yang ingin dipakai juga *kernel* RBF, linear, dan polinolian sehingga tidak perlu melakukan perubahan *kernel*. Untuk nilai C akan dilakukan dalam lima nilai. Semakin besar nilai C maka proses semakin lama, tetapi nilai C yang semakin besar dipercaya dapat menghasilkan hasil yang lebih akurat. Nilai C yang semakin kecil maka model yang dihasilkan akan lebih *smooth*, maka dari itu digunakanlah proses dengan nilai C yang berbeda.

4.13. Perncangan Pembuatan Model

Pada tahap terakhir ini akan dibuat model. Data yang dimiliki sejak awal belum dipisahkan mana untuk *data training* mana yang untuk *data testing*. Maka dari itu, sebelum dibuat model, data dibagi menjadi dua, yaitu 70% untuk *data training*, dan 30% untuk *data testing*. Pemisahan ini dilakukan agar terbagi menjadi dua bagian data. Pembagian 70% dan 30% merupakan pembagian yang paling sering dilakukan. *Data training* yang sudah dipisahkan akan dibagi lagi menjadi 70% dan 30%, 30% tersebut untuk validasi. Pemisahan tersebut dilakukan agar pengujian lebih akurat dan dapat tervalidasi. Setelah dipisahkan dan sudah terbentuknya vektor, maka model pun siap dibuat. Setelah keluar model, akan dilihat akurasinya, yang nantnya akan menentukan metode SVM cocok atau tidak dalam pembentukan model untuk jenis data seperti ini. Setelah pembuatan model ini, maka akan dinilai dengan akurasi. Akurasi sendiri merupakan suatu perhitungan yang mengukur seberapa deka tantara model yang dibuat dengan data actual yang ada. Adapun rumus akurasi sebagai berikut (2.12):

$$Accuracy = \frac{Jumlah\ Prediksi\ yang\ Benar}{Total\ Keseluruhan\ Data} \qquad (2.12)$$

BAB V
IMPLEMENTASI PERMODELAN

Pada bab ini akan dijelaskan terkait implementasi dari desain yang telah dilakukan sesuai dengan metode pengerjaan yang telah ada dibab sebelumnya. Bagian ini akan menjelaskan mengenai lingkungan implementasi, pembuatan code, pembuatan model, dan pengujian model.

5.1. Persiapan Perangkat

Pada tahap ini perangkat keras dan perangkat lunak yang akan dipakai akan didata dan dipersiapkan terlebih dahulu. Sebelum melakukan pengerjaan, terdapat perangkat keras dan perangkat lunak yang harus dipersiapkan. Perangkat keras yang disiapkan ada dua, yaitu laptop dan USB. Laptop merupakan laptop pribadi mahasiswa, begitu pula dengan USB.

Tabel 5.1. Spesifikasi Perangkat Keras

Perangkat Keras	Komponen	Spesifikasi
Laptop Asus Zenbook 13 UX333F	Prosesor	Intel® Core™ i5-8265U 1.6GHz (Turbo up to 3.9GHz)
	Ram	8 GB
	Sistem Operasi	Windows 10 (64bit)
USB Toshiba	Kapasitas	16 GB

Selain perangkat keras, terdapat perangkat lunak yang perlu didiapkan juga. Beberapa perangkat lunak itu adalah editor kode pemrograman, bahasa pemrograman, dan *library*.

Tabel 5.2. Tabel Perangkat Lunak

Perangkat Lunak	Yang Digunakan
Editor Kode	Jupyter Notebook
Bahasa Pemrograman	Phyton
Library	• Numpy • Pandas • Matplotlib

	• Nltk • Sklearn • Gensim • Copy • String • Warnings • Collections • Twitterscraper

Bahasa pemrograman yang digunakan untuk pengerjaan adalah Python, hal itu didasari dari bahas pemrograman ini memiliki *library* yang cukup banyak dan lengkap, sehingga tidak perlu membangun kode pemrograman murni dari awal, dapat melakukan pemanggilan *library* saja. Selain itu digunakan juga beberapa library seperti yang telah dicantumkan pada Tabel 5.2.

5.2. Crawling Data

Pada tahap ini akan dilakukan pengumpulan data dengan *crawling*. Hal tersebut dilakukan karena data yang didapat tidaklah cukup, karena data yang tersedia terlalu lama, dan rentan waktunya relatif terlalu sebentar pula. Hal tersebut dapat menyebabkan hasil yang keluar nantinya tidak relevan, sehingga kurang sesuai. Untuk melengkapi data, maka dilakukan tahap *crawling*.

```
import twitterscraper
twitterscraper "@KAI121" --output databaru.csv --begindate 2017-1-1 --enddate 2019-10-30
```

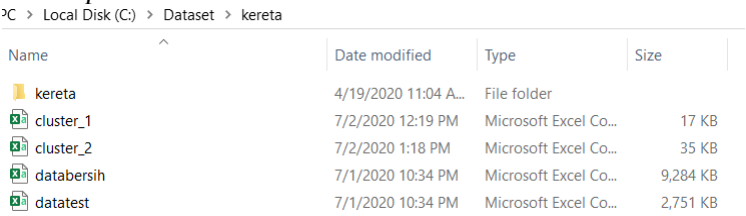
Kode 5.1. Kode Crawling Data

Kode 5.1 menunjukkan bahwa akan ada *crawling* data mulai dari awal 2017, sampai dengan akhir 2019. Dalam melakukan *crawling*, perlu mengimport library 'twitterscraper' pada python. Hal tersebut dilakukan karena data yang dapat di-*crawling* terbatas, dan juga membutuhkan waktu yang sangat lama. Data sampai 2019 pun sudah cukup relevan, sehingga data yang didapat disatukan dengan data yang sudah disediakan. Proses crawling ini dilakukan untuk melengkapi data, karena bila menggunakan data yang sudah ada yaitu

hanya sampai 2016, maka hasilnya akan menjadi kurang maksimal dikarenakan data yang ada tersebut sudah cukup lama sehingga kurang relevan lagi terhadap situasi saat ini.

5.3. Pengumplan Data

Pada tahap ini akan dilakukan pengumpulan data. Sebelum melakukan pengerjaan, maka dataset yang sudah ada perlu dikumpulkan dan dicek kembali kelengkapannya. Data yang dibutuhkan ada empat *file*, *file* tersebut adalah daftar tweet KAI sebanyak 81.918 baris. Tweet KAI tersebut didapat dari data yang memang sudah ada, ditambah dengan data hasill crawling sampai dengan oktober 2019. Data yang dipakai adalah data dengan format csv. Data yang hendak dipakai dimasukkan kedalam satu folder. Data berupa *tweet* dan juga barisan kata-kata. Selain itu, ada daftar kata- kata bernada positif, negatif, dan *stop words*.



Gambar 5.1. File yang Dibutuhkan

Data tersebut disatukan kedalam satu projek, sehingga memudahkan proses pengolahan. Selain dataset, dilakukan juga proses pemasukkan data berupa daftar kata klaster positif, klaster negatif, dan *stop word*. Tetapi file yang memuat data tersebut baru akan terbentuk saat ditengah- tengah pengerjaan, setelah proses *clustering*. Data tersebut berupa cluster yang dianggap mewakili tiga jenis sentiment, yaitu sentimen positif, sentiment negatif, dan sentiment netral. Sehingga dapat dikatakan pahwa proses pemasukkan file akan terjadi dua kali, yaitu diawal pengerjaan dan ditengah pengerjaan.

5.4. Pemasukan Data dan Library

Pada tahap ini akan dimasukkan library yang akan digunakan dalam melakukan proses mulai dari pemasukan data sampai melakukan analisa sentimen.

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
from random import random, sample
from scipy import stats
from scipy.sparse import hstack
import re
import copy
import string
import warnings
import collections

# Matplot
import matplotlib.pyplot as plt
%matplotlib inline

# Scikit-learn
from sklearn.model_selection import train_test_split, StratifiedKFold, cross_val_score, GridSearchCV
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import confusion_matrix, accuracy_score, roc_auc_score, recall_score, precision_score, make_scorer, auc
from sklearn.manifold import TSNE
from sklearn.feature_extraction.text import TfidfVectorizer, CountVectorizer
from sklearn.pipeline import make_pipeline, Pipeline
from sklearn.cluster import KMeans
from sklearn.svm import SVC

# nltk
import nltk
from nltk.corpus import stopwords
from nltk.stem import SnowballStemmer, WordNetLemmatizer
from nltk.tokenize import word_tokenize, TweetTokenizer
from nltk import ngrams

# Word2vec
import gensim



count_vectorizer = CountVectorizer(ngram_range=(1,2))
warnings.filterwarnings('ignore')
nltk.download('stopwords')
```

Kode 5.2 Kode Import Library yang Digunakan

```
# Import Dataset
bersih = pd.read_csv('C:/Dataset/kereta/databersih.csv', encoding = "ISO-8859-1") #data training
testdata = pd.read_csv('C:/Dataset/kereta/datatest.csv', encoding = "ISO-8859-1") #data testing

# Import Sentimen
stop = pd.read_csv('C:/Dataset/kereta/stopwordsID.csv', names=['stopwordsID'])
positive = pd.read_csv('C:/Dataset/kereta/cluster_1.csv', names=['positivewords'])
negative = pd.read_csv('C:/Dataset/kereta/cluster_2.csv', names=['negativewords'])

stop = stop.values
```

Kode 5.3 Kode Import Data

Pada Kode 5.2 dapat dilihat bahwa ada beberapa *library* yang akan digunakan dalam melakukan pengerjaan ini. Library yang di-import cukup banyak agar menghindari import library ditengah- tengah pengerjaan. Selain library ada juga proses import dataset sseperti yang dapat dilihat pada Kode 5.3, dilakukan juga proses pemasukkan data berupa daftar kata positif, negatif, dan *stop word*. Import dataset dilakukan diawal pengerjaan, tetapi untuk import klaster negatif dan klaster

positif dilakukan ditengah- tengah pengerjaan, setelah proses *clustering*.

5.5. Praproses Data

Pada proses ini akan dilakukan pada Jupyter Notebook, dengan bahasa pemrograman Phyton. Tahap ini akan berisi terkait proses pengolahan data sehingga data mentah dapt diolah menjadi data yang bersih.

5.4.1. Penghilangan Data Duplikat

Pertama yang dilakukan adalah menghapus data duplikat. Prosesnya tersebut dengan melakukan pengecekan terhadap data yang ada, bila baris data memiliki kesamaan dengan baris data lainnya, maka salah satunya harus dihapus, sementara yang satu lagi tetap dipertahankan. Hal itu dilakukan agar tidak semua data duplikat terhapus.

```
bersih.drop_duplicates(keep=False,inplace=True)
bersih.reset_index(drop=True, inplace=True)
```

Kode 5.4 Kode Penghapusan Duplikat

Setelah itu akan dilakukan seperti Kode 5.4 diatas untuk menghapus secara permanen baris yang memiliki kesamaan. Setelah dilakukan Langkah ini, ditemukan cukup banyak data yang terduplikasi.

5.4.2. Penyamarataan Huruf

Dalam melakukan analisa sentimen, huruf yang digunakan sangat berpengaruh dan menentukan. Huruf kecil dan huruf besar sangat mempengaruhi. Maka dari itu, dataset kumpulan tweet dirubah huruf besarnya menjadi huruf kecil. Hal tersebut dilakukan agar dapat dinilai dengan mudah bila dicocokan dengan daftar kata- kata baik positif, negatif, ataupun *stop-word*. Maka di python akan dimasukkan kode yaitu `bersih['isi'] = bersih.isi.str.lower()`. Setelah melakukan Langkah ini, maka seluruh huruf yang berada dalam kolom isi yang mengandung *tweet*- *tweet* yang ada akan berubah menjadi huruf kecil semua. Pengubahan kata menjadi *lowercase* ini disebut *casefolding*. Proses in memanfaatkan fungsi lower dari

pandas, sehingga seluruh *tweet* dapat diubah menjadi *lowercase.*

5.4.3. Pembersihan Dataset

Pada tahap ini akan dilakukan pembersihan dataset dilakukan dengan menghilangkan berbagai tanda baca, *link*, *mention*, *hashtag*, serta kata kertait KAI121. Hal itu dilakukan karena bagian- bagian yang tidak penting tersebut perlu dihilangkan.

```
def replace_mention(string):
    tokens = ['' if token.startswith('@') else token for token in string.split()]
    return ' '.join(tokens)

def replace_URL(string):
    tokens = ['' if 'http' in token else token for token in string.split()]
    return ' '.join(tokens)

def replace_hastag(string):
    tokens = ['' if token.startswith('#') else token for token in string.split()]
    return ' '.join(tokens)

pattern = "@\S+|https?:\S+|http?:\S|[^A-Za-z0-9]+"
def clean_text(x):
    x = str(x)
    x = x.replace('kai121', ' ')
    x = x.replace('KAI121',' ')
    x = x.replace('-KAI121',' ')
    x = x.replace('KAI-121',' ')
    x = x.replace('-kai121',' ')
    x = x.replace('kai-121',' ')
    x = x.replace(pattern, ' ')
    return x
```

```
bersih["isi"] = bersih["isi"].apply(lambda x: clean_text(x))
bersih["isi"] = bersih["isi"].apply(lambda string: replace_mention(string))
bersih["isi"] = bersih["isi"].apply(lambda string: replace_URL(string))
bersih["isi"] = bersih["isi"].apply(lambda string: replace_hastag(string))
```

Kode 5.5 Kode Tokenisasi

Kode 5.5 diatas merupakan kode untuk menghilangkan baik *link, hastag*, dan *mention*. Setelah dihapus, maka akan diganti dengan kata spasi, sehingga tidak menyambung ke kata- kata lain yang bisa membuat kata- kata lainnya tidak bisa diproses. *Library* yang digunakan pada proses ini adalah *library* re (regular expression) dan pandas. Pertama yang dilakukan memanggil fungsi replace_mention, fungsi ini akan menghapus *mention* yang ada pada data *tweet*. Kedua, memanggil fungsi replace_URL yang akan menghapus URL yang ada pada data *tweet*. Ketiga, mamanggil fungsi replace_hastag untuk menghapus hastagnya.

5.4.4. Penghilangan Missing Value

Terdapat beberapa baris yang tidak ada isi *tweet*. Baris tersebut perlu dihapus. Selain itu ada baris yang ketika dimasukkan ke sistem tidak dapat terdefinisi. Berbagai cara yang dilakukan untuk menghilangkan *missing value*. Cara- cara tersebut adalah, menggantinya dengan data terbanyak, menggantinya dengan rata- rata, atau menghapusnya secara permanen. Dari beberapa solusi yang ditawarkan, dipilihkan penghapusan data secara permanen. Hal ini dipilih karena tidak ada rat- rata *tweet* yang paling sering keluar, dan juga tidak ada kalimat yang paling banyak keluar. Untuk menghindari presepsi yang kurang pendasar, maka diputuskan dengan menghilangkan data secara permanen. Ada pun data-data tersebut, akan dihilangkan.

Gambar 5.2. Data yang Dihilangkan

Bila ada baris seperti dataset diatas, akan langsung dihilangkan secara permanen. Untuk menghilangkan data tersebut dibuatlah kode perintah.

```
#cleansing missing value
bersih = bersih[bersih.isi != '#REF!']
bersih = bersih[bersih.isi != '#NAME?']
bersih.reset_index(drop=True, inplace=True)
```

Kode 5.6. Kode Pembersihan Data

Dengan menjalankan Kode 5.6, maka *missing value* akan menghilang permanen, sehingga data lebih siap lagi untuk diproses.

5.5. Tokenisasi

Pada tahap ini akan dilakukan tahap tokenisasi. Tujuan dari tokenisasi sendiri adalah untuk memecah kalimat kedalam kata- kata. Kata- kata ini dipisahkan agar kita dapat melakukan ekstraksi makna dari sebuah *tweet* yang ada. Ekstraksi ini akan menunjukkan makna dari kata perkata. Dengan pemecahan ini, maka dapat dilakukan pengecekan kata satu persatu. Kode

perintah `testdata['clean'] = testdata['isi'].apply(lambda x: [item for item in x if item not in stops])` merupakan kode perintah untuk melakukan ekstraksi tersebut yang diharapkan akan mempermudah dalam melakukan pemberian nilai nantinya.

	no	asal	jenis	kategori	sub_kategori	subjek	isi	clean
0	6279559.0	Social media	Informasi	Informasi Produk PT KAI	Informasi Kereta Penumpang	Informasi Kereta Ekonomi	[sy, berniat, beli, tiket, untk, 6, org, mau, ...	[berniat, beli, untk, 6, no, kursi, 6, penumpa...
1	6279444.0	Social media	Informasi	Informasi Produk PT KAI	Informasi Kereta Penumpang	Informasi Kereta Ekonomi	[posisi, tempat, duduk, di, kereta, progo, bia...	[posisi, duduk, progo, ya, min, contoh, 16a17a...
2	6279523.0	Social media	Informasi	Informasi Layanan PT KAI	Informasi Layanan & Fasilitas Stasiun	Informasi Pelayanan Stasiun	[loket, pembatalan, tiket, buka, sampai, jam, ...	[loket, pembatalan, buka, jam, stasiun, pasar...
3	6279471.0	Social media	Informasi	Informasi Produk PT KAI	Informasi Kereta Penumpang	Informasi Kereta Ekonomi	[min, mau, tanya, kalo, ka, bogowonto, itu, se...	[min, kalo, ka, bogowonto, seatnya, 22, 33, ya...
4	6279864.0	Social media	Informasi	Informasi Ticketing	Informasi Prosedur Ticketing	Informasi Pemesanan Tiket Rombongan	[cara, pemesanan, tiket, kereta, api, rombonga...	[pemesanan, api, rombongan, yaa]
5	5778589.0	Social media	Informasi	Informasi Rute & Jadwal	Informasi Rute & Jadwal Kereta Lokal	Informasi Rute & Jadwal Kereta Lokal	[min, info, tentang, kereta, api, lokal, surab...	[min, info, api, lokal, surabaya, akun]
6	5778649.0	Social media	Informasi	Informasi Tarif	Informasi Tarif Kereta Lokal	Informasi Tarif Kereta Lokal	[ka, lokal, purwakarta, yg, ac, berapa, ya, ha...	[ka, lokal, purwakarta, ac, ya, harga, tiketnya]
7	5778698.0	Social media	Informasi	Informasi Rute & Jadwal	Informasi Rute & Jadwal Kereta Lokal	Informasi Rute & Jadwal Kereta Lokal	[min, ka, lokal, solopurwokerto, ada, nggak, yal...	[min, ka, lokal, solopurwokerto, nggak, yal

Gambar 5.3. Tweet yang sudah di Array

Gambar 5.3 diatas merupakan contoh dari teks yang sudah melewati tokenisasi dan dimasukkan kedalam *array* untuk memudahkan melakukan pengecekan makna satu persatu kata, mulai dari penyocokan dengan daftar kata positif, negatif, maupun *stop-words*.

5.6. Pembuatan Vektor

Pada tahap ini akan dilakukan proses pembuatan data vektor terlebih dahulu. Pembuatan data vektor sendiri dilakukan untuk mengganti data bertipe teks menjadi vektor agar lebih mempercepat dalam melakukan proses- proses selanjutnya. Pada tahap ini, vektorisasi akan seperti mengoptimasi sehingga langkah selanjutnya menjadi lebih cepat, bila langkah ini diabaikan dan tidak dilakukan maka harus dlakukan pengecekantipe data pada satu persatu item yang cukup melelahkan.

```
#vektorisasi
desc = bersih['tweet'].values
vectorizer = TfidfVectorizer(ngram_range=(1, 1), tokenizer=tokenizer.tokenize)

vectorizer.fit(bersih.tweet)
train_vectorized = vectorizer.transform(bersih['tweet'])
X = vectorizer.fit_transform(desc)
words = vectorizer.get_feature_names()
```

Kode 5.7. Kode Vektorisasi

Kode 5.5 ini dibuat tepat sebelum proses pembentukan klaster dan juga SVM. Vektorisasi digunakan untuk mempercepat kode

Python tanpa menggunakan loop. Menggunakan fungsi seperti itu dapat membantu dalam meminimalkan waktu berjalan kode secara efisien. Setelah proses ini data teks yang telah diubah menjadi tipe data vektor ini dapat langsung dimasukkan dalam clustering maupun pembuatan model. Data yang telah menjadi vektor, yang telah dibagi menjadi dua yaitu 70% untuk *data training* dan 30% untuk *data testing,* akan langsung diolah masuk kedalam tahap selanjutnya.

5.7. Pembembentukan Klaster

Pada tahap ini akan dilakukan pengelompokkan *tweet* ini akan dibuat kedalam tiga kelompok agar bisa dipisah- pisah menjadi positif, negatif, dan netral. Pengelompokkan tersebut dibuat dengan cara kluster menggunakan *k-means.* Data yang akan di-*input* dalam proses ini adalah data vektor yang telah dihasilkan terlebih dahulu oleh TfidfVektorizer. Pengelompokkan tersebut dibuat dengan cara melakukan *clustering* pada seluruh kata yang terdapat dalam *tweet* yang ada didalam dataset.

Tweet yang ada akan diklaster menjadi tiga klaster yaitu positif, negative, dan netral.

```
kmeans = KMeans(n_clusters = 3, n_init = 5, n_jobs = -1)
kmeans.fit(X)
```

```
common_words = kmeans.cluster_centers_.argsort()[:,-1:-11:-1]
for num, centroid in enumerate(common_words):
    print(str(num) + ' : ' + ', '.join(words[word] for word in centroid))
```

```
# Memasukkan kata kedalam csv
common_words = kmeans.cluster_centers_.argsort()

cluster_1 = []
cluster_2 = []
cluster_3 = []


for num, centroid in enumerate(common_words):
    #print(str(num) + ' : ' + ', '.join(words[word] for word in centroid))
    if num == 0:
        for word in centroid:
            cluster_1.append(words[word])
        cluster_1 = np.array(cluster_1)
        cluster1 = pd.DataFrame({
            "kata": cluster_1
        })
        cluster1.to_csv("cluster1.csv", index=False)
    elif num == 1:
        for word in centroid:
            cluster_2.append(words[word])
        cluster_2 = np.array(cluster_2)
        cluster2 = pd.DataFrame({
            "kata": cluster_2
        })
        cluster2.to_csv("cluster2.csv", index=False)
    elif num == 2:
        for word in centroid:
            cluster_3.append(words[word])
        cluster_3 = np.array(cluster_3)
        cluster3 = pd.DataFrame({
            "kata": cluster_3
        })
        cluster3.to_csv("cluster3.csv", index=False)
```

Kode 5.8. Kode Clustering K-Means

Kode 5.6 diatas membagi kedalam tiga cluster. Sehingga akan muncul *cluster* negative, positif, dan netral, baik dalam pembagian secara metode *k-means*. Setelah dilakukan clustering, maka kata yang masuk akan diperiksa kembali dan dipilih mana klaster yang paling sesuai dengan sentimen yang dimiliki.

```
from sklearn.metrics import silhouette_score
```

```
silhouette_score(X, labels=kmeans.predict(X))
```

Kode 5.9. Kode Silhouette Coefficient

Setelah itu akan dilakukan penilaian terhadap cluster tersebut menggunakan silhouette coefficient seperti yang tertera pada Kode 5.9. Nilai hasil silhouette coefficient terletak pada kisaran nilai -1 sampai +1. Jika nilai silhouette coefficient mendekati nilai 1, maka semakin pengelompokan data dalam satu cluster dapat dikatakan cukup baik. Tetapi nilai silhouette coefficient jika mendekati nilai -1, maka dapat dikatakan bahwa pengelompokan data didalam satu cluster buruk sekali.

5.8. Pengecekan Kata yang Paling Sering Muncul

Pada tahap ini dilakukan pengecekan terhadap kata yang paling sering keluar. Pengecekan itu dilakukan untuk melihat mana kata yang paling sering keluar, sehingga bisa diketahui apa saja yang paling sering dibicarakan atau dikomentari oleh pengguna layanan perusahaan ini di laman *twitter* perusahaan.

```
kata = {}
for arr in bersih.isi:
    for word in arr:
        if word not in kata:
            kata[word] = 1
        else:
            kata[word] += 1
```

Kode 5.10. Kode Pengecekan Banyaknya Kemunculan Kata

Kode 5.10 perintah yang ada diatas dibuat agar kata- kata seluruhnya masuk kedalam suatu *array*. Setelah itu kata-kata baru akan dicek, terkait berapa kali kata tersebut keluar pada dataset yang dimiliki. Kata yang keluar tersebut akan dihitung berapa kali kemunculannya dan akan dituliskan berapa kali kata tersebut muncul. Hal itu dilakukan agar kata benar- benar dapat mudah dilihat berapa kali kemunculannya. Kata yang paling sering muncul, kemungkinan adalah yang paling sering dibahas dan paling diperhatikan oleh para pengguna layanan.

5.9. Pemberian Skor pada Tweet

Pada tahap ini akan dilakukan penilaian terhadap *tweet* yang ada, penilaian itu diawali dengan pemberian nilai berkata lalu ditotalkan.

5.9.1. Pemberian Nilai

Pemberian nilai diawali dengan mengumpulkan kata 'nggak', 'tidak', 'ga', 'tak', 'tdk', 'enggak' dalam satu *array*. Penyatuan kata- kata tersebut dalam suatu *array* adalah ada hubungannya dengan kata sinikal. Dalam penilaian ini, setiap kata positif akan dinilai +1, untuk kata negatif akan dinilai -1, sementara netral akan dinilai 0, sehingga terdapat tiga kelompok.

```
#Scoring
pnw = ['nggak','tidak','ga','tak','tdk','enggak']
rekap =[]
for i in range(0, len(bersih.clean)):
    score = 0
    for kata in bersih.clean[i]:
        if kata in positiveW:
            score +=1
        elif kata in negativeW:
            score -=1

        if kata in pnw:
            index = bersih.clean[i].index(kata)
            kata_sifat = bersih.clean[index+1]

            if kata_sifat in positiveW:
                score -= 2
            elif kata_sifat in negativeW:
                score += 2
```

Kode 5.11. Kode Pemberian Nilai

Kode 5.11 diatas merupakan perintah untuk melakukan penilaian. Kata 'nggak', 'tidak', 'ga', 'tak', 'tdk', 'enggak' disatukan dalam *array* pnw[]. Lalu jika kata tersebut ada dalam daftar kata positif, akan diberi nilai +1, bila adanya di negatif, akan diberi nilai -1. Terkait kata sinikal tersebut, biasanya kata sinikal merupakan kata positif yang didepannya ada kata yang merupakan anggota *array* pnw[]. Maka dari itu, bila kata tersebut ada di *array* positif maka akan diberi -2, hal itu dikarenakan kata tersebut sebelumnya sudah mendapat nilai +1 diperhitungan sebelumnya, sehingga bila sinikal nilainya jadi -1, sementara bila kata yang merupakan anggota pnw bersandingan dengan kata dalam daftar kata negatif, maka kata maknanya adalah positif, sehingga diberi +2, karena pasti

sudah dinilai -1 diperhitungan sebelumnya, bila diberi +2, nilai akhirnya menjadi +1.

5.9.2. Total Nilai Akhir

Nilai yang sudah didapat akan ditotalkan, sehingga setiap tweet dapat memiliki satu nilai, nilai ini didapat dari penjumlahan semua nilai yang didapat dari kata- kata yang ada dalam satu *tweet*.

```
    rekap.append(score)

def listToString(s):
    str1 = " "
    return (str1.join(s))

#Total Score
bersih['skor'] = pd.Series(rekap)
```

Kode 5.12. Kode Pentotalan Nilai

Nilai yang sudah ada akan dimasukkan kedalam kolom baru yang bernama ‘skor’. Kolom ini akan berisi totalan nilai. Kolom ini merupakan hasil dari penilaian terhadap kolom ‘clean’ yang berisi tweet yang telah ditokenisasi sehingga kata perkata saling terpisah.

5.10. Pemberian Label

Pada tahap ini akan dilakukan pemberian label dilakukan dengan cara pelabelan sesuai dengan nilai yang dimiliki, sehingga dapat dilihat secara jelas banyak dari *tweet* yang bernada positif, negatif maupun netral. Pemberian label dilakukan dengan memberikan label ‘negative’ untuk nilai -1 kebawah, lalu 0 untuk ‘neutral’, dan label ‘positive’ untuk nilai +1 keatas. Kode untuk membuat pelabelan tersebut adalah `bersih['sentimen'] = ['negative' if x < 0 else 'positive' if x > 0 else 'neutral' for x in bersih.skor]`. Hal tersebut dilakukan untuk melihat berapa banyak data terdapat tweet positif, negatif, dan netral dan juga untuk proses klasifikasi. Selain itu dengan adanya pemberian label tersebut maka akan memudahkan untuk tahap SVM, karena SVM

sendiri membutuhkan data berlabel untuk dapat melakukan pengerjaannya.

5.11. Penyetelan Parameter

Penyetelan parameter dilakukan dengan sangat sederhana. Penyetelan parameter hanya dengan mengubah nilai C yang dimiliki. Nilai C diganti menjadi C=5.0, C=10.0, C=15.0, C=20.0, C=25.0. Hal tersebut dilakukan hanya untuk membuktikan bahwa perubahan suatu nilai dapat mempengaruhi hasil. Nilai C yang berubah berpengaruh pada proses yang membutuhkan waktu yang semakin panjang pula. C= 1.0 adalah C yang paling umum dan paling sering digunakan. Bila data berjumlah banyak biasanya disarankan untuk menggunakan nilai C yang lebih kecil dari pada 1. Nilai C sendiri sensitive terhadap *noise* maka pada data ini tidak perlu dilakukan pengecilan nilai C, karena noise yang dihasilkan tidak signifikan. Selain itu untuk gamma disini dipakai 'scale'.

5.12. Pembuatan Model

Pembuatan model dibuat dengan melakukan pelatihan pada data *training* yang sudah dibagi sebanyak 70%. Dilanjutkan dengan melakukan prediksi model menggunakan data testing. Secara langsung, *kernel* yang dipakai adalah *kernel* RBF, linear, dan polinonial. Untuk data input, digunakan data vektor, hal itu dikarenakan data vektor dapat mempercepat. Selain pada hal ini digunakan SVC, karena bila dilihat dari persentase perbandingan data, data yang ada termasuk *unbalance data*, karena tweet bernilai negatif termasuk sangat seikit, sementara untuk yang positif dan netral termasuk banyak dan mendominasi. SVM sendiri memiliki nilai C. SVC sendiri tidak terlalu sensitive terkait perubahan parameter, sehingga dapat diartikan bahwa SVC mudah menyesuaikan nilai dan model.

```
X_train, X_test, y_train, y_test = train_test_split(train_vectorized, y, test_size=0.3, random_state = 123, stratify=y)
```

Kode 5.13. Kode Pemisahan Data

```
kernels = ['linear', 'poly', 'rbf']
C_params = [5, 10, 15, 20, 25]

for k in kernels:
    for c in C_params:
        model = SVC(kernel=k, C=c)
        model.fit(X_train, y_train)
        y_pred = model.predict(X_test)
        accuracy = accuracy_score(y_test, y_pred)
        print("kernel=", k,'C=', c, "akurasi=", accuracy)
```

Kode 5.14. Kode Pembuatan Model SVM

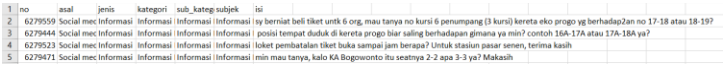
Setelah menjalankan kode program seperti yang ada diatas, maka akan keluar hasil akurasinya, walau perlu waktu yang lebih lama.

BAB VI
HASIL DAN PEMBAHASAN

Pada bab ini akan berisi tentang hasil dari penelitian beserta pembahasan dari implementasi yang sudah di lakukan sesuai dengan metode penelitian. Hasil yang akan dibahas di bab ini adalah mengenai Praproses data, Klasifikasi *Support Vector Machine* dan Analisa Sentimen.

6.1. Hasil Data

Pada hasil praproses data yang sebelum nya merupakan data tabel yang tersusun dengan baik. Data awalnya memiliki 7 kolom. Kolom tersebut teriri dari nomor, asal, jenis, kategori, sub_kategori, subjek dan isi.



	no	asal	jenis	kategori	sub_kategi	subjek	isi
2	6279559	Social mec	Informasi	Informasi	Informasi	Informasi	sy berniat beli tiket untk 6 org, mau tanya no kursi 6 penumpang (3 kursi) kereta eko progo yg berhadap2an no 17-18 atau 18-19?
3	6279444	Social mec	Informasi	Informasi	Informasi	Informasi	posisi tempat duduk di kereta progo biar saling berhadapan gimana ya min? contoh 16A-17A atau 17A-18A ya?
4	6279523	Social mec	Informasi	Informasi	Informasi	Informasi	loket pembatalan tiket buka sampai jam berapa? Untuk stasiun pasar senen, terima kasih
5	6279471	Social mec	Informasi	Informasi	Informasi	Informasi	min mau tanya, kalo KA Bogowonto itu seatnya 2-2 apa 3-3 ya? Makasih

Gambar 6.1. Gambar Data Berupa Tabel Awal

Dari data yang tersedia, hanya kolom isi yang digunakan. Kolom isi tersebut digunakan untuk diidentifikasi kata-katanya. Selain kolom isi, kolom lainnya dihilangkan. Hal tersebut dilakukan karna hanya kolom isi yang dibutuhkan. Untuk data yang digunakan adalah data dari 2016 – 2019 bulan oktober.

Setelah melakukan pengahapusan terhadap data yang tidak diinginkan, data dibagi dua, 70% dan 30%. Hal tersebut untuk *testing* akhir.

6.2. Hasil Penghapusan Duplikasi Dataset

Pada hasil yang didapat, terdapat beberapa data yang merupakan data duplikat. Data duplikat ini harus dihapus karena data yang merupakan duplikat dapat mempengaruhi nilai dan model. Sehingga, perlu adanya pengecekan dan penghapusan.

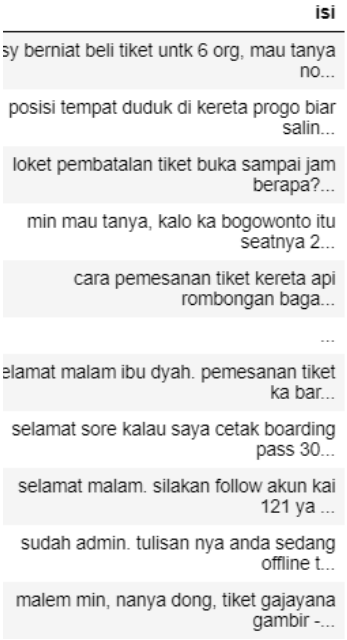
Tabel 6.1. Tabel Data Duplikat

Isi
Lucky Lutdiansyah ?-DeLuckyPiero 1h1 hour ago - KAI121 gitu ajah jawabannya? Ga ada tanggapan lebih lanjut? Wadooooh ini kereta ekonomi ajah kalah panasnya!!! ??? 6:26 PM - 22 Aug 2015 åá
Lucky Lutdiansyah ?-DeLuckyPiero 1h1 hour ago - KAI121 gitu ajah jawabannya? Ga ada tanggapan lebih lanjut? Wadooooh ini kereta ekonomi ajah kalah panasnya!!! ??? 6:26 PM - 22 Aug 2015 åá
Dynar Manggiasih ?-dynarmanggis 1h1 hour ago -kai121 ini tawang jaya tujuan jakarta kenapa tumben telat banget ya? jam segini baru nyampe cikarang 9:07 PM - 24 Aug 2015 åá
Dynar Manggiasih ?-dynarmanggis 1h1 hour ago -kai121 ini tawang jaya tujuan jakarta kenapa tumben telat banget ya? jam segini baru nyampe cikarang 9:07 PM - 24 Aug 2015 åá

Gambar diatas merupakan data duplikan yang didapat pada pengecekan data mentahan, data ini perlu dihapus sehingga model yang dihasilkan dapat lebih akurat.

6.3. Hasil Pembersihan Data

Hasil proses pembersihan data ini dilakukan dengan tujuan untuk menghilangkan karakter yang tidak memiliki makna dan juga meminimalisir kata- kata yang bermakna sama tetapi dianggap berbeda hanya karena adanya tanda baca.



isi
sy berniat beli tiket untk 6 org, mau tanya no...
posisi tempat duduk di kereta progo biar salin...
loket pembatalan tiket buka sampai jam berapa?...
min mau tanya, kalo ka bogowonto itu seatnya 2...
cara pemesanan tiket kereta api rombongan baga...
...
elamat malam ibu dyah. pemesanan tiket ka bar...
selamat sore kalau saya cetak boarding pass 30...
selamat malam. silakan follow akun kai 121 ya ...
sudah admin. tulisan nya anda sedang offline t...
malem min, nanya dong, tiket gajayana gambir -...

Gambar 6.2. Gambar Data Bersih

Selain membersihkan dari tanda baca, *tweet* yang ada dibersihkan juga dari *mention, link*, dan juga *hastag.* Hal itu dilakukan karena bisa kata- kata Bersatu dengan *mention* dan juga *hashtag,* maka kata tersebut dianggap kata berbeda dari kata yang sama tetapi tidak melekat pada *hashtag.* Selain itu, terdapat juga pembersihan jam *posting*. Hal itu harus dihapus karena tidak bermakna untuk proses selanjutnya.

6.4. Hasil Tokenisasi

Data yang sudah dibersihkan, dimasukkan kedalam *array* yang ada. *Array* tersebut dibuat untuk setiap baris data yang telah dibersihan. Gunanya pembuatan *array* ini adalah agar kata perkata dapat terpisah.

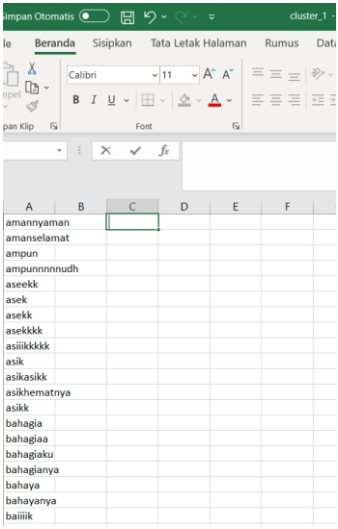
clean
[berniat, beli, untk, 6, no, kursi, 6, penumpa...
[posisi, duduk, progo, ya, min, contoh, 16a17a...
[loket, pembatalan, buka, jam, stasiun, pasar,...
[min, kalo, ka, bogowonto, seatnya, 22, 33, ya...
[pemesanan, api, rombongan, yaa]
[min, info, api, lokal, surabaya, akun]
[ka, lokal, purwakarta, ac, ya, harga, tiketnya]

Gambar 6.3. Gambar Data Bersih yang Dimasukkan Array

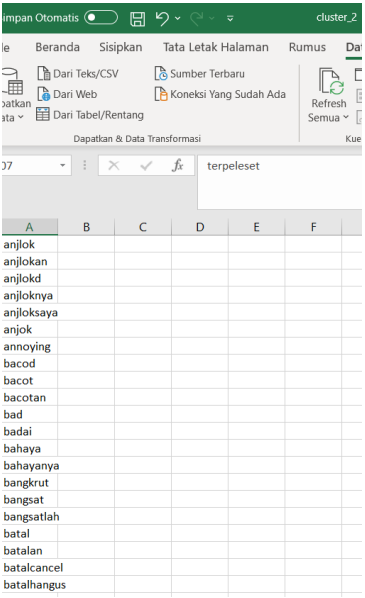
Kata yang terpisah ini akan memudahkan sistem ketika mencocokkan dengan daftar kata yang telah ada. Kata- kata yang merupakan *stopwords*, dihilangkan dari dalam *array*, sehingga hal ini dapat lebih lagi mempermudah proses pengindentifikasian kata- kata, sehingga maknanya dapat disimpulkan dengan baik.

6.5. Hasil Clustering

Pada hasil pengelompokkan, pengelompokkan yang dilakukan dengan menggunakan bantuan *clustering*. Setiap kata akan masuk kedalam kelompok- kelompok tertentu. Kelompok tersebut ada untuk dikeluarkan datanya dalam bentuk daftar kata pada file *excel* dengan formal csv untuk membantu melakukan penilaian.



Gambar 6.4. Gambar Hasil Clustering Cluster Positif

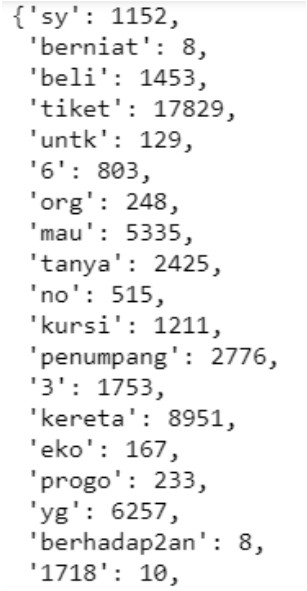


Gambar 6.5. Gambar Hasil Clustering Cluster Negatif

Tahap *clustering* ini dianggap dapat membantu melakukan pelabelan. Tahap ini banyak membantu walau juga menghasilkan kesalahan dalam pengelompokkan kata karena banyak kata yang salah diberikan pada kluster. Maka dari itu, *clustering* menggunakan *k-means* ini perlu dilakukan pemeriksaan kata- katanya secara manual agar dapat dipastikan tidak ada kata yang salah masuk klaster. Selain itu kata yang masuk kedalam klaster hanya akan dihapus bila ada yang salah, tetapi tidak dilakukan tukar menukar kata dari satu klaster ke klaster lainnya untuk melakukan perbaikan klaster. Penentuan klaster positif dan negatif ditentukan dengan melakukan *scanning* pada anggota klaster. Lalu ditemukan bahwa anggota klaster yang ke 2 lebih mengandung banyal kata negatif. Klaster ini sebelumnya akan dievaluasi dengan sebuah metode. Metode tersebut adalah silhouette coefficient, yang merupakan gabungan dari metode cohesion yang berfungsi untuk mengukur kedekatan relasi antara objek dalam sebuah cluster, dan metode separation yang mengukur seberapa jauh sebuah cluster terpisah dengan cluster lain. Lalu didapatlah hasil yaitu 0.0086.Bukan hasil yang baik, tetapi tidak terlalu buruk juga, karena hasil terburuknya adalah -1, dan hasil terbaiknya adalah +1. Sehingga nilai dapat dikatakan cukup baik.

6.6. Hasil Pencarian Kata yang Keluar

Dari seluruh *tweet* yang ada, banyak sekali hal yang dikomentari baik yang sering maupun yang jarang sekali. Untuk mengetahui kata tersebut, harus dilihat kata apa saja yang sering muncul dalam *tweet* tersebut.



Gambar 6.6. Gambar Banyak Kata yang Keluar

Dapat dilihat bahwa banyak kata- kata berulang sampai ribuan kali, kata- kata yang berulang tersebut dapat dilihat bahwa ada kata- kata yang mengandung makna positif dan negatif, ada juga yang tidak mengandung makna, tetapi dari sini dapat dilihat bagian atau kata apa saja yang paling sering disebut dan hal terkait apa saja yang paling sering diperhatikan oleh pelanggan. Kata yang paling sering keluar adalah terkait tiket dan keterlambatan. Maka artinya, pengguna layanan menaruh perhatian pada hal tersebut. Sehingga ha tersebut yang perlu diperhatikan, karena hal tersebut bila tidak diperhatikan dapat mengurangi tingkat kepuasan sehingga pelanggan mungkin akan berpindah kelain transportasi.

6.7. Hasil Pemberian Nilai

Data yang sudah dibersihkan diberi penilaian. Pemberian nilai ini dilakukan dengan mencocokan kata- kata yang telah dipisah kedalam array dan dipisah lagi satu persatu data pada kolom

'clean'. Kata dalam kolom ini dicocokan satu persatu dengan kata yang ada. Setelah dicocokan, data diberi nilai. Pemberian nilai dilakukan dengan mencocokan kata, bila kata tersebut terdapat didaftar kata positif, maka kata tersebut akan diberi nilai +1, bila kata tersebut berada di kolom negatif, maka akan diberi -1. Pemberian nilai itu akan menghasilkan *output* seperti pada yang ada pada gambar. Output dari proses ini dapat dilihat pada tabel skor yang berisi angka-angka hasil penilaian.

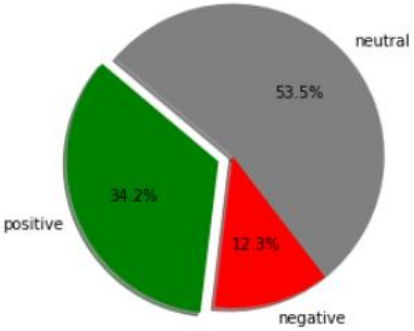
isi	clean	tweet	skor	sentimen
[sy, berniat, beli, tiket, untk, 6, org, mau, ...	[berniat, beli, untk, 6, no, kursi, 6, penumpa...	berniat beli untk no kursi penumpang kursi ...	0	neutral
[posisi, tempat, duduk, di, kereta, progo, bia...	[posisi, duduk, progo, ya, min, contoh, 16a17a...	posisi duduk progo ya min contoh aa aa ya	3	positive
[loket, pembatalan, tiket, buka, sampai, jam, ...	[loket, pembatalan, buka, jam, stasiun, pasar,...	loket pembatalan buka jam stasiun pasar senen ...	0	neutral
[min, mau, tanya, kalo, ka, bogowonto, itu, se...	[min, kalo, ka, bogowonto, seatnya, 22, 33, ya...	min kalo ka bogowonto seatnya ya makasih	1	positive
[cara, pemesanan, tiket, kereta, api, rombonga...	[pemesanan, api, rombongan, yaa]	pemesanan api rombongan yaa	0	neutral

Gambar 6.7. Gambar Tweet yang Sudah Dinilai

Nilai yang ada berasal dari penilaian dari kata- kata yang sudah dimasukkan ketika tahap pembentukan klaster. Nilai ini yang akan menentukan *tweet* tersebut akan masuk klaster yang mana diantara tiga klaster yang adaa.

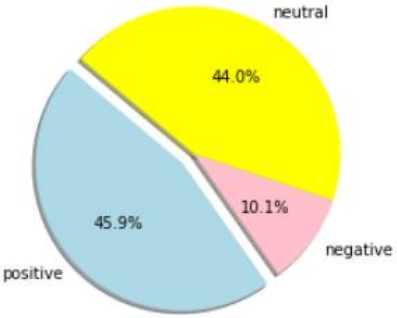
6.8. Hasil Persentase Nilai

Setelah melakukan *clustering* dengan *k-means* dan melakukan pelabelan. Maka dilakukanlan tahapan untuk melihat persentase data pada tweet yang sudah ada.



Gambar 6.8. Persentase Kelas Data Training

Dari *data training*, terdapat 34,2% *tweet* bernada positif, 12.3% *tweet* bernada negatif, dan 53.5% *tweet* bernada netral.



Gambar 6.9. Persentase Kelas Data Testing

Dari *data testing* dapat dilihat bahwa 45,9% *tweet* bernada positif, 10,1% *tweet* bernada negatif, dan 44,0% tweet bernada netral. Dapat dikatakan bahwa pola *data training* maupun *testing* hampir sama, karena positif dan netral yang banyaknya hamper sama dikedua dataset sementara negatif yang paling sedikit.

```
#sentimen bersih
sentimen_counts = bersih.sentimen.value_counts()
print(sentimen_counts)
```

```
neutral     31334
positive    20045
negative     7221
```

Gambar 6.10. Banyak Anggota Kluster Data Training

Dari segi jumlah data pada *data training* dapat diketahui bahwa jumlah terbanyak adalah untuk data netral sebanyak 31334 *tweet*, lalu diikuti oleh data positif sebanyak 20045 *tweet*, dan data negatif sebanyak 7221 *tweet*.

```
#sentimen testdata
sentimen_counts = testdata.sentimen.value_counts()
print(sentimen_counts)
```

```
positive    12726
neutral     12217
negative     2802
```

Gambar 6.11. Banyak Anggota Kluster Data Training

Dari segi jumlah data pada *data testing* dapat diketahui bahwa jumlah terbanyak adalah untuk data positif sebanyak 12726 *tweet*, lalu diikuti oleh data netral sebanyak 12217 *tweet*, dan data negative sebanyak 2802 *tweet*.

6.9. Hasil Pemberian Pelabelan

Setelah melakukan penilaian, maka langkah selanjutnya adalah melakukan pemberian label. Langkah ini dilakukan untuk memberikan label pada setiap *tweet* yang ada. Label yang ada ini akan berguna untuk proses SVM sendiri.

Tabel 6.2. Tabel Hasil Pelabelan

Tweet	Sentimen
sy berniat beli tiket untk 6 org, mau tanya no kursi 6 penumpang (3 kursi) kereta eko progo yg berhadap2an no 17-18 atau 18-19?	Netral
Twitter agus kurniawan ?-aguskur70937319 7m7 minutes ago -	Negatif

KAI121 KA Eksekutif Bangunkarta mogok di sekitar Bekasi, udah sejam gak bisa jalan. Kerusakan ada di lokomotiv. OMG 4:15 PM - 31 Jul 2015 åá	
Udah puas jalan-jalannya	Positif

Setiap tweet yang sudah dikelompokkan, dilakukan pengecekan ulang, apakah memang benar sudah terkelompokkan dengan baik atau tidak. Lalu ditemukanlah seperti tabel diatas.

6.10. Hasil Skenario

Setelah dilakukan pelabelan, maka dicarilah model yang tepat untuk memodelkan data. Pada permodelan, sempat mengganti parameter yaitu untuk nilai C. Nilai C yang digunakan adalah C= 5.0, C= 10.0, C= 15.0, C= 20.0, C= 25.0. Untuk kernel yang digunakan adalah RBF, linear, dan polinonial. Penggunaan tiga buah kernel dan lima nilai C yang berbeda dianggap dapat dibandingkan agar mendapat hassil yang terbaik. Data sendiri sebenarnya dibagi menjadi dua. Data bagian pertama dilakukan untuk melakukan training dan validasi. Pembagiannya adalah 70% dan 30%. Untuk pembagian data tersebut didapatlah data seperti yang ada pada tabel.

Tabel 6.3. Tabel Akurasi Data Training

	C	Akurasi
Linear	5.0	0.96308
	10.0	0.96331
	15.0	0.96313
	20.0	0.96313
	25.0	0.96336

Polynonial	5.0	0.75073
	10.0	0.74129
	15.0	0.74135
	20.0	0.74129
	25.0	0.74129
RBF	5.0	0.93196
	10.0	0.93185
	15.0	0.93185
	20.0	0.93191
	25.0	0.93185

Dari tabel diatas dapat diketahui bahwa terdapat perbedaan bila diberikan nilai C yang berbeda. Perbedaan dari C dan kernel memang tidak terlalu besar tetapi cukup signifikan. Sehingga dapat dikatakan bahwa kernel yang paling tepat dipakai dalam model ini adalah kernel linerar, dengan nilai C terbaik pada 15 kali percobaan adalah 25. Model yang dibuat akan dicoba langsung dengan *data testing.*

Tabel 6.4. Tabel Akurasi Data Testing

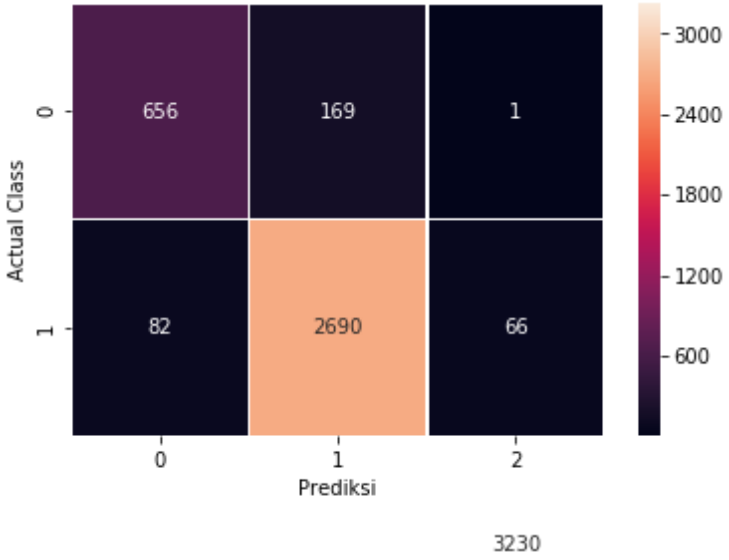
Kernel	C	Akurasi
Linear	25.0	0.93078

Setelah digunakan untuk *data testing*, dapat dilihat bahwa perbedaan nilai C sangat membawa pengaruh pada data testing.

Hal tersebut dapat dilihat dari hasil model yang dihasilkan oleh data testing. Perbedaan cukup signifikan. Hal ini membuktikan bahwa pengaruh nilai C cukup kuat, sehingga menghasilkan hasil seperti tabel diatas.

6.11. Hasil Confusion Matrix

Setelah melakukan percobaan menggunakan SVM, maka dihasilkanlah confusion matrix. Confusion matrix ini dihasilkan dari SVM dengan hasil yang terbaik yaitu SVM dengan kernel linear dengan nilai C-25.0. Nilai tersebut menghasilkan akurasi 96%.



Gambar diatas merupakan confusion matix yang telah dibahas sebelumnya.

Halaman ini sengaja dikosongkan

BAB VII
KESIMPULAN DAN SARAN

Pada bab ini akan dibahas tentang kesimpulan dan saran yang didapatkan dari proses pelaksaaan penelitian tugas akhir ini untuk perkembangan yang lebih baik lagi.

7.1. Kesimpulan

Setelah melakukan beberapa prose dan langkah pengerjaan, maka didapat beberapa kesimpulan dari tugas akhir ini:

1. Tugas Akhir ini telah berhasil mengimplementasikan model pengklasifikasi teks ber¬basis support vector machine (SVM) yang dikombinasikan dengan algoritma klasterisasi k-means untuk melakukan analisis sentimen unggahan twitter dari kastemer di sebuah perusahaan transportasi. Algoritma k-means digunakan untuk memberikan pelabelan data pelatihan secara semi-otomatis, sedang pengklasifikasi SVM digunakan untuk mengelompokkan jenis sentimen (positif, negatif, atau netral) dari semua unggahan twitter yang digunakan sebagai data tes.Komentar negatif ditemukan, tetapi tidak terlalu banyak persentasenya yaitu 12,3%. Sehingga dapat dikatakan layanan mengalami perbaikan, dan semakin bertambahnya umur perusahaan, komentar negatif semakin berkurang. Tetapi hal itu mengharuskan perusahaan lebih meningkatkan pelayanan kepada pelanggan, misalnnya dengan membuat call center 24 jam atau daftar pertanyaan dan jawaban yang paling sering ditanyakan.
2. Hasil uji coba implementasi SVM dengan menggunakan tiga jenis kernel berbeda (linier, polinomial, dan RBF) dan dengan lima variasi parameter C yang berbeda untuk setiap jenis kernal mmemberikan hasil akurasi klasifikasi jenis sentimen rata-rata berturut-turut sebesar 96.33%, 93,19%, dan 75,07% untuk jenis kernel linier, RBF, dan polinomial.
3. Dari hasil analisis sentimen ditemukan komentar negatif sebesar 12,3%. Hal ini mengidikasikan bahwa layanan yang

diberikan oleh perusahaan relatif baik. Selain itu, hasil analisis menunjukkan bahwa aspek yang paling sering dikomentari berkaitan dengan tiket, jadwal, dan keterlambatan keberangkatan maupun kedatangan kereta api. Untuk ini, perusahaan harus memberikan perhatian lebih terhdaap ketiga aspek tersebut.

7.2. Saran

1. Perlu dilakukan percobaan dengan metode clustering yang lainnya, sehingga mendapat hasil yang lebih baik lagi.
2. Perlu dilakukan pembobotan kata sebelum dijadikan vektor dengan metode selain TF-IDF
3. Rentan nilai untuk kelompok netral perlu diperluas. Hal itu perlu dilakukan agar tweet yang bernada netral lebih banyak lagi dapat tertangkap oleh model.
4. Data yang kurang seimbang membuat banyak tweet yang tercampur antara kelompok netral dan positif.
5. Perlu percobaan dengan metode klasifikasi yang lain.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

[1] A. N. N. N. B. M. Petric, "SVM-based Models for Mobile Users' Initial Position Determination," *The Journal of Navigation*, 2014.

[2] A. S. a. G. Gonzalez, "Sentiment Analysis in Twitter using Centroids, Clusters, and Sentiment Lexicons," *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 2016.

[3] M. G. O. S. Siordia, "Sentiment Analysis for Twitter: TASS 2015," *ISSN*, 2015.

[4] N. F. F. d. S. E. R. H. E. R. H. J. Luiz F. S. Coletta, "Combining Classification and Clustering for Tweet Sentiment Analysis," *Brazilian Conference on Intelligent Systems*, 2014.

[5] Twitter, "Twitter," Twitter, [Online]. Available: https://about.twitter.com/. [Accessed 6 9 2019].

[6] P. Bhatia, "Principles and Practical Techniques Paperback," in *Data Mining and Data Warehousing*, Cambridge, Cambridge, 2019, pp. 110-111.

[7] C. M. Bishop, "Pattern Recognition and Machine Learning (Information Science and Statistics)," New York, 2016.

[8] E. S. Y. S. Damarsari Cahyo Wilogo, "Mendeteksi Spammers Di Twitter Dengan Svm Classifier," *Sematic*, 2018.

[9] D. G. A. R. K. N. G. Ravi Kumar, "An Efficient Feature Selection System to Integrating SVM with Genetic Algorithm for Large Medical Datasets," *Sematic*, 2014.

[10] R. S. Bayu Yudha Pratama, "Personality classification based on Twitter text using Naive Bayes, KNN and

SVM," in *International Conference on Data and Software Engineering (ICoDSE)*, Yogyakarta, 2015.

[11] J. I. S. N. S. Vijayarani, "Preprocessing Techniques for Text Mining-An Overview Dr," *Computer Science*, 2015.

[12] A. R. A. ,. D. S. M. ,. R. S. L. ,. W. D. a. M. A. R. C Slamet, "Automated Text Summarization for Indonesian Article Using Vector Space Model," in *The 2nd Annual Applied Science and Engineering Conference* , Napoli, 2017.

[13] S. P. M. A. S. S. E. D. T. J. B. G. K. K. Mehdi Allahyari, "A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques," no. Cornell University, 2017.

[14] W. L. M.-M. H. Sung-Sam Hong, "The Feature Selection Method based on Genetic Algorithm for Efficient of Text Clustering and Text Classification," 2015.

[15] A. T. K. Laith Mohammad Abualigah, "Unsupervised text feature selection technique based on hybrid particle swarm optimization algorithm with genetic operators for the text clustering," *The Journal of Supercomputing*, no. Springer, 2017.

[16] S. A. T. W. D. C. L. ArmanKhadjeh Nassirtoussia, "Text mining of news-headlines for FOREX market prediction: A Multi-layer Dimension Reduction Algorithm with semantics and sentiment," no. Science DIrect, 2015.

[17] S. K. Munyaradzi W.Nyadzayo, "The antecedents of customer loyalty: A moderated mediation model of customer relationship management quality and brand image," *Journal of Retailing and Consumer Services* , no. Elsevier, 2016.

[18] A. S. a. G. Gonzalez, "Sentiment Analysis in Twitter using Centroids, Clusters, and Sentiment Lexicons," Vols. Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), 2016.

[19] M. G. Oscar S. Siordia, "Sentiment Analysis for Twitter: TASS 2015," *ISSN*, 2015.

Halaman ini sengaja dikosongkan

BIODATA PENULIS

Penulis bernama Helena Angelita Depari, lahir di Singkawang pada tanggal 14 Juli 1998, yang merupakan anak perta dari dua bersaudara. Penulis menempuh Pendidikan formal di beberapa sekolah yaitu: SDK 3 BPK Penabur Jakarta (2005-2011), SMPK1 BPK Penabur Jakarta (2011-2013), dan SMAK 3 BPK Penabur (2013-2016). Penulis melanjutkan Pendidikan sarjana di Departemen Sistem Informasi Fakultas Teknologi Informasi dan Komunikasi (FTIK) Institut Teknologi Sepuluh Nopember (ITS) pada tahun 2016 sebagai mahasiswi dengar nomor nrp 05211640000062.

Selama menjadi mahasiswa Penulis aktif mengikuti beberapa kegiatan perlombaan. Kegiatan tersbut adalah menjadi finalis Arkavidia ITS, Datavidia 2020, finalis JOINTS UGM 2020, serta menjadi finalis FINDIT UGM 2020. Selain itu, penulis juga melakukan magang di Bank btpn pada bulan Juli 2019 – Agustus 2019.

Untuk mendapatkan gelar S.Kom (Sarjana Komputer), penulis mengambil topik penelitian tugas akhir klasifikasi pada laboratorium Rekayasa Data dan Intelegensi Bisnis. Untuk kepentingan penelitian, Penulis dapat dihubungi melalui email helenaangelita1407@gmail.com