

**ANALISIS KLASIFIKASI SENTIMEN *REVIEW* APLIKASI
E-TICKETING MENGGUNAKAN METODE *SUPPORT*
VECTOR MACHINE DAN ASOSIASI**

(Studi Kasus : *Review* Aplikasi KAI Access dan Tiket.com pada *Google Play*)

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana

Program Studi Statistika



Norma Fikria

14 611 131

JURUSAN STATISTIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS ISLAM INDONESIA

YOGYAKARTA

2018

HALAMAN PERSETUJUAN PEMBIMBING

TUGAS AKHIR

Judul : Analisis Klasifikasi Sentimen *Review* Aplikasi *E-Ticketing* Menggunakan Metode *Support Vector Machine* dan Asosiasi. (Studi Kasus : *Review* Aplikasi *KAI Access* dan *Tiket.com* pada *Google Play*)

Nama Mahasiswa : Norma Fikria

Nomor Mahasiswa : 14 611 131

TUGAS AKHIR INI TELAH DIPERIKSA DAN DISETUJUI UNTUK
DIUJIKAN

Yogyakarta, 17 April 2018

Pembimbing



(Tuti Purwaningsih, S.Stat., M.Si.)

HALAMAN PENGESAHAN

TUGAS AKHIR

ANALISIS KLASIFIKASI SENTIMEN *REVIEW* APLIKASI

E-TICKETING MENGGUNAKAN METODE *SUPPORT VECTOR*

MACHINE DAN ASOSIASI

(Studi Kasus : *Review* Aplikasi KAI Access dan Tiket.com pada *Google Play*)



Nama Mahasiswa : Norma Fikria

Nomor Mahasiswa : 14 611 131

TUGAS AKHIR INI TELAH DIUJIKAN
PADA TANGGAL 15 MEI 2018

Nama Penguji

Tanda Tangan

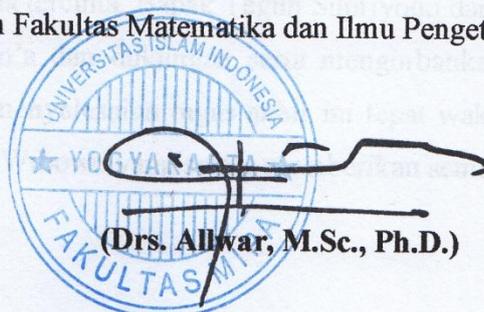
1. Ir. Ali Parkhan, M.T.

2. Ayundyah Kesumawati, S.Si., M.Si.

3. Tuti Purwaningsih, S.Stat., M.Si.

Mengetahui,

Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



(Drs. Allwar, M.Sc., Ph.D.)

KATA PENGANTAR

Bismillahirrahmaanirrahiim.

Assalamu'alaikum Wr. Wb.

Alhamdulillah, puji dan syukur kehadirat Allah SWT yang telah melimpahkan rahmat, taufiq dan hidayah-Nya, sehingga penulis dapat menyusun dan menyelesaikan tugas akhir yang berjudul “Analisis Klasifikasi Sentimen *Review* Aplikasi *E-Ticketing* Menggunakan Metode *Support Vector Machine* dan Asosiasi. (Studi Kasus : *Review* Aplikasi KAI Access dan Tiket.com pada *Google Play*)” sebagai salah satu syarat yang harus dipenuhi dalam menyelesaikan jenjang strata satu di Jurusan Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Islam Indonesia. Shalawat serta salam semoga selalu tercurah kepada Nabi Muhammad SAW beserta sahabat dan para pengikutnya yang senantiasa menjaga keimanan hingga akhir hayatnya.

Penyelesaian tugas akhir ini tidak terlepas dari dukungan, arahan, dan bimbingan, dan bantuan baik materi maupun non materi dari berbagai pihak, oleh karena itu penulis ingin menghaturkan ucapan terimakasih kepada :

1. Drs. Allwar. M.Sc, Ph.D. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam.
2. Dr. RB. Fajriya Hakim, M.Si. selaku Ketua Program Studi Statistika Universitas Islam Indonesia beserta jajarannya.
3. Ibu Tuti Purwaningsih, S.Stat.,M.Si. selaku dosen pembimbing yang selalu sabar dalam membimbing dalam penyelesaian tugas akhir ini.
4. Seluruh staf pengajar Program Statistika Universitas Islam Indonesia yang telah memberikan ilmu yang luar biasa banyang dan Insya Allah bermanfaat.
5. Kedua orang tua tercinta, Bapak Teguh Supriyono dan Ibu Rohyati yang telah memberikan do'a dan dukungan serta mengorbankan banyak hal sehingga penulis dapat menyelesaikan tugas akhir ini tepat waktu, serta adik tersayang Satriyo Agung Wibowo yang selalu memberikan semangat dan motivasi.

6. Teman terbaik Laras Mega Kalpita, Rosita Dewi dan Dwi Luviana yang selalu menemani pergi ke Indomart dan selalu memberikan semangat serta dukungannya.
7. Teman KKN unit 125 angkatan 56 : Nurul, Mas Siswan, dan Fathli yang memberikan semangat untuk menyelesaikan tugas akhir ini.
8. Sahabat terbaik : Irawan, Indra, Leli, Agus, dan Lita yang selalu menyebarkan semangat dan menemani video call saat pengerjaan tugas akhir ini.
9. Pihak – pihak lain yang mungkin belum disebutkan, yang telah membantu dalam penyusunan tugas akhir ini.

Semoga dukungan dan bantuan dari semua pihak senantiasa mendapat balasan dari Allah SWT. Penulis menyadari bahwa dalam tugas akhir ini masih terdapat banyak salah dan kekurangan karena keterbatasan pengetahuan dan kemampuan penulis. Oleh karenanya, kritik dan saran yang bersifat membangun sangat diharapkan demi sempurnanya penyusunan tugas akhir ini. Semoga tugas akhir ini bermanfaat bagi semua pihak. Aamiin.

Wassalamu'alaikum wr wb.

Yogyakarta, 15 Mei 2018

Penulis

DAFTAR ISI

HALAMAN PERSETUJUAN PEMBIMBING	ii
HALAMAN PENGESAHAN	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN	xii
PERNYATAAN	xiii
INTISARI	xiv
ABSTRACT	xv
BAB I	1
PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	4
1.4 Jenis Penelitian dan Metode Analisis	4
1.5 Tujuan Penelitian	4
1.6 Manfaat Penelitian	5
BAB II	6
TINJAUAN PUSTAKA	6
2.1 Transportasi	6
2.2 Profil PT Kereta Api Indonesia	6
2.3 <i>E-ticketing</i>	7
2.3.1 KAI Access	7
2.3.2 Tiket.com	8
2.4 Google Play	9
2.5 Penelitian Terdahulu	9
BAB III	14
LANDASAN TEORI	14
3.1 <i>Online Review</i>	14
3.2 Analisis Deskriptif	14
3.3 Data Mining dan Klasifikasi	15

3.4	Analisis Sentimen.....	17
3.5	<i>Text Mining</i>	18
	3.5.1 <i>Spelling Normalization</i>	18
	3.5.2 <i>Case Folding</i>	18
	3.5.3 <i>Tokenizing</i>	18
	3.5.4 <i>Filtering</i>	19
3.6	Term Frequency-Inverse Document Frequency (TF-IDF).....	19
3.7	Algoritma Prediksi Support Vector Machine.....	22
	3.7.1 <i>SVM Linearly Separable Data</i>	22
	3.7.2 <i>SVM Nonlinearly Separable Data</i>	25
3.8	Evaluasi Sistem Klasifikasi.....	29
	3.8.1 K-Fold Cross Validation.....	29
	3.8.2 Precision.....	30
	3.8.3 Recall	30
	3.8.4 F-Measure	30
	3.8.5 Accuracy	31
3.9	Word Cloud.....	31
3.10	Asosiasi Kata.....	31
3.11	Diagram Sebab Akibat (<i>Cause and Effect Diagram</i>).....	32
BAB IV	33
METODOLOGI PENELITIAN	33
4.1	Populasi dan Sampel	33
4.2	Variabel dan Definisi Operasional Variabel	33
4.3	Jenis dan Sumber Data	33
4.4	Metode Analisis Data	33
4.5	Tahapan Penelitian	34
BAB V	35
PEMBAHASAN	35
5.1	Analisis Deskriptif.....	35
5.2	Teks Preprocessing.....	38
	5.2.1 <i>Spelling Normalization</i>	39
	5.2.2 <i>Case Folding</i>	39
	5.2.3 <i>Tokenizing</i>	40
	5.2.4 <i>Filtering</i>	41

5.3	Representasi Model	42
5.3.1	Perhitungan Skor Sentimen	42
5.3.2	Pelabelan Kelas Sentimen.....	42
5.4	Klasifikasi Sentimen	44
5.4.1	Data Latih dan Data Uji	44
5.4.2	Klasifikasi <i>Support Vector Machine</i>	45
5.5	Visualisasi dan Asosiasi	47
5.5.1	KAI Access	48
5.5.2	Tiket.com	56
BAB VI	63
PENUTUP	63
6.1	Kesimpulan.....	63
6.2	Saran.....	64
DAFTAR PUSTAKA	65
LAMPIRAN	70

DAFTAR TABEL

Tabel 2. 1 : Perbandingan Penelitian Terdahulu	12
Tabel 3. 1 : Perhitungan TF-IDF.....	22
Tabel 3. 2 : Confusion Matrix	30
Tabel 4. 1 : Variabel Definisi Operasional.....	33
Tabel 5. 1 : Perhitungan Skor Sentimen.....	42
Tabel 5. 2 : Hasil Pelabelan Kelas Sentimen	43
Tabel 5. 3 : Hasil Pelabelan Kelas Sentimen Tanpa Netral	43
Tabel 5. 4 : Pembuatan Data Uji dan Data Latih	44
Tabel 5. 5 : Perbandingan Accuracy Metode Kernel	45
Tabel 5. 6 : Perbandingan Nilai Accuracy Machine Learning.....	46
Tabel 5. 7 : Confusion Matrix	46
Tabel 5. 8 : Asosiasi Kata Positif KAI Access.....	49
Tabel 5. 9 : Asosiasi Kata Negatif KAI Access	53
Tabel 5. 10 Rencana Pemecahan Masalah Aplikasi KAI Access	55
Tabel 5. 11 : Asosiasi Kata Positif Tiket.com.....	58
Tabel 5. 12 : Asosiasi Kata Negatif Tiket.com	60
Tabel 5. 13 Rencana pemecahan Masalah Aplikasi Tiket.com	61

DAFTAR GAMBAR

Gambar 1. 1 : Grafik Volume Penumpang Kereta Api	1
Gambar 2. 1 : Logo PT Kereta Api Indonesia	6
Gambar 2. 2 : Logo Google Play	9
Gambar 2. 3 : Logo Aplikasi KAI Access	7
Gambar 2. 4 : Logo Aplikasi Tiket.com	8
Gambar 3. 1 : Tahap-Tahap Knowledge Discovery from Data (KDD).....	16
Gambar 3. 2 : Alternatif Bidang Pemisah dan Bidang Pemisah Terbaik.....	23
Gambar 3. 3 : Soft Margin Hyperplane.....	26
Gambar 3. 4 : Transformasi dari vektor input ke feature space	27
Gambar 3. 5 : Transformasi Data Tidak Dapat Dipisahkan Secara Linier	27
Gambar 3. 6 : Ilustrasi 5-Fold Cross Validation	29
Gambar 4. 1 : Diagram Alir Penelitian	34
Gambar 5. 1 : Perbandingan Jumlah Review	35
Gambar 5. 2 Jumlah Review KAI Access Feb-Mar.....	36
Gambar 5. 3 : Perbandingan Rating Aplikasi.....	37
Gambar 5. 4 : Jumlah Review Aplikasi KAI Access	37
Gambar 5. 5 : Jumlah Review Aplikasi Tiket.com	38
Gambar 5. 6 : Proses Spelling Normalization	39
Gambar 5. 7 : Proses Case Folding	40
Gambar 5. 8 : Proses Tokenizing	40
Gambar 5. 9 : Proses Filtering.....	41
Gambar 5. 10 : Kata Frekuensi Terbanyak dari Kelas Positif KAI Access	48

Gambar 5. 11 : Word Cloud Kelas Positif KAI Access.....	49
Gambar 5. 12 : Kata Frekuensi Terbanyak dari Kelas Negatif KAI Access.....	51
Gambar 5. 13 : Word Cloud Kelas Negatif KAI Access	51
Gambar 5. 14 Diagram Sebab Akibat KAI Access.....	54
Gambar 5. 15 : Kata Frekuensi Terbanyak dari Kelas Positif Tiket.com	57
Gambar 5. 16 : Word Cloud Kelas Positif Tiket.com.....	57
Gambar 5. 17 : Kata Frekuensi Terbanyak dari Kelas Negatif Tiket.com.....	59
Gambar 5. 18 : Word Cloud Kelas Negatif Tiket.com	59
Gambar 5. 19 Diagram Sebab Akibat Tiket.com.....	61

DAFTAR LAMPIRAN

Lampiran 1 Script Python Web Scrapping KAI Access

Lampiran 2 Script Python Web Scrapping Tiket.com

Lampiran 3 Script R Preprocessing Data dengan Text Mining KAI Access

Lampiran 4 Script R Preprocessing Data dengan Text Mining Tiket.com

Lampiran 5 Script R Pelabelan Kelas Sentimen KAI Access

Lampiran 6 Script R Pelabelan Kelas Sentimen Tiket.com

Lampiran 7 Script R Klasifikasi menggunakan SVM KAI Access

Lampiran 8 Script R Klasifikasi menggunakan SVM Tiket.com

Lampiran 9 Script R Visualisasi dan Asosiasi Kata KAI Access

Lampiran 10 Script R Visualisasi dan Asosiasi Kata Tiket.com

Lampiran 11 Stopwords

PERNYATAAN

Dengan ini saya menyatakan bahwa dalam Tugas Akhir ini tidak terdapat karya yang sebelumnya diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 15 Mei 2018



Norma Fikria

**ANALISIS KLASIFIKASI SENTIMEN *REVIEW* APLIKASI
E-TICKETING MENGGUNAKAN METODE *SUPPORT VECTOR
MACHINE* DAN ASOSIASI**

(Studi Kasus : *Review* Aplikasi KAI Access dan Tiket.com pada *Google Play*)

Norma Fikria

Program Studi Statistika Fakultas matematika dan Ilmu Pengetahuan Alam

Universitas Islam Indonesia

INTISARI

Review pada google play merupakan salah satu fitur yang digunakan untuk memberikan suatu penilaian terhadap suatu aplikasi. KAI Access dan Tiket.com merupakan aplikasi e-ticketing pada google play yang berguna dalam proses penjualan tiket secara online dengan memanfaatkan teknologi internet. Namun review pada aplikasi e-ticketing tersebut hanya berupa teks tanpa arti tertentu, untuk itu diperlukan analisis yang dapat mengklasifikasikan review sebagai sentimen pengguna. Dalam penelitian ini diterapkan tahap preprocessing untuk mengolah data mentah suatu review agar dapat dibuat sebuah model yang merepresentasikan data yang diolah menggunakan algoritma klasifikasi Support Vector Machine (SVM). Klasifikasi menggunakan algoritma SVM menghasilkan akurasi klasifikasi sebesar 89.60% pada review KAI Access dan 84.68% pada review Tiket.com. Hasil klasifikasi menunjukan perbedaan sentimen pada kedua aplikasi, pada KAI Access review yang masuk pada kelas sentimen negatif lebih tinggi jumlahnya sedangkan pada aplikasi Tiket.com review yang masuk pada kelas sentimen positif lebih tinggi jumlahnya. Secara umum sentimen positif pada aplikasi KAI Access berkaitan dengan adanya ngopi gratis, tampilan dan fitur aplikasi menarik sedangkan pada aplikasi Tiket.com berkaitan dengan banyaknya discount dan promo yang menjadikan harga tiket menjadi murah. Sentimen negatif pada aplikasi KAI Access berkaitan dengan errornya aplikasi setelah di update dan susah nya mendapatkan tiket kereta untuk lebaran sedangkan pada aplikasi Tiket.com berkaitan dengan susah nya refund dan banyaknya iklan pada aplikasi tersebut.

Kata Kunci : *Google Play, E-Ticketing, Support Vector Machine, KAI Access, Tiket.com*

CLASSIFICATION SENTIMENT ANALYSIS REVIEW E-TICKETING APPLICATION USING SUPPORT VECTOR MACHINE AND ASSOCIATION METHOD

(Case Study : *Review KAI Access and Tiket.com Application on Google Play*)

Norma Fikria

Statistics Department, Faculty of Mathematics and Science

Islamic University of Indonesia

ABSTRACT

Review on google play is one of fitur which is used for giving a rating toward an application. KAI acces and Tiket.com are e-ticketing application on google play which are useful on online ticket sale which takes advantage by using internet. But, review on e-ticketing application is only text without any certain meaning, for that it is needed an analysis which can classify review as user sentimen. On this research, it is applied preprocessing step for processing an original review in order to made a model which can presented data which is processed by using algoritma classification support vector machine (SVM). With the result of an accurate classification in the amount of 89,60% on KAI acces review and 84,68% on tiket.com review. The result in classification showed the differentiation on both of two applications. On KAI Acces review which is entering pisitive sentimen class is having more higher number. Generally, positive sentimen on KAI process is related to "ngopi gratis" appearance and fitur of application on KAI acces is interesting, meanwhile tiket.com application is related to quantities of discount and promotion which is making the tiket is cheap. Negative sentimen on KAI acces application is related to the error of application after updated and it is difficult to get train ticket for lebaran, meanwhile on tiket.com application is related to the difficult to refund and there are a lot of advertisement on that application.

Keyword : *Google Play, E-Ticketing, Support Vector Machine, KAI Access, Tiket.com*

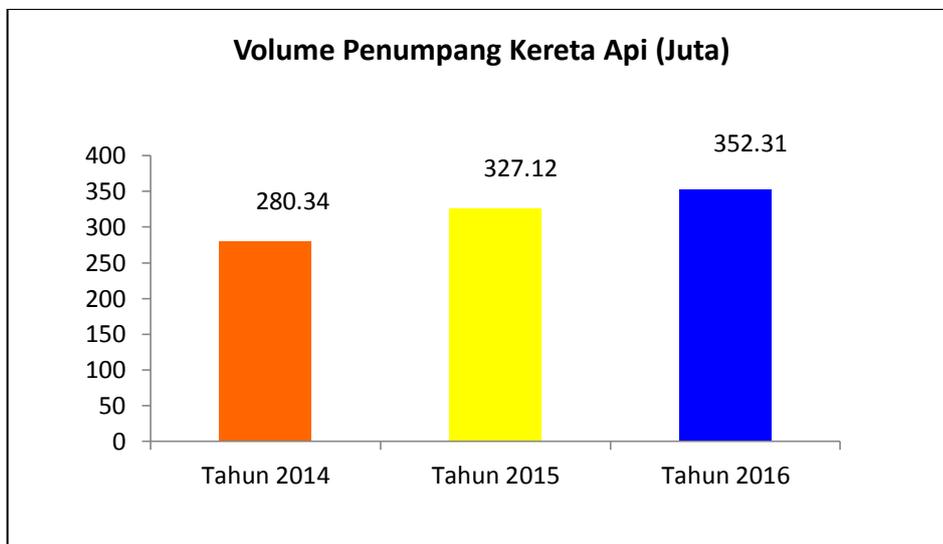
BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Pertumbuhan pengguna internet di Indonesia mengalami peningkatan setiap tahunnya, menurut hasil survei Asosiasi Jasa Internet Indonesia (APJII), pengguna internet di Indonesia pada tahun 1998 sampai dengan tahun 2017 selalu mengalami kenaikan, khususnya pada tahun 2017 naik sebesar 7.33% dari tahun 2016. Naiknya angka pengguna internet mendorong perkembangan digital yang berpengaruh terhadap semua bidang kehidupan sehari-hari tak terkecuali pada bidang transportasi di Indonesia.

Saat ini sektor transportasi di Indonesia telah mulai memanfaatkan internet guna menunjang sistem transportasi yang lebih berkualitas. Bangsa yang maju adalah bangsa yang memiliki sistem transportasi yang berkualitas. Sistem transportasi yang berkualitas diperlukan untuk menunjang berkembangnya sektor lain guna mendorong perkembangan daerah tertinggal (Adisasmita, 2011).



Gambar 1. 1 : Grafik Volume Penumpang Kereta Api (Juta)

PT. Kereta Api Indonesia (Persero) juga selalu melakukan inovasi guna meningkatkan kualitas pelayanan bagi pengguna transportasi kereta api di Indonesia. Hal ini terbukti dengan naiknya volume penumpang kereta api di Indonesia dalam tiga tahun terakhir, pada tahun 2016 penumpang kereta api di

Indonesia mencapai 352.31 juta penumpang. Jumlah tersebut naik 7.1 % dari tahun sebelumnya. Peningkatan jumlah penumpang kereta api tak membuat pihak PT. KAI persero puas dengan pencapaiannya. Hal ini dibuktikan dengan adanya inovasi baru seperti penerapan sistem *check-in* dan *boarding pass* serta peluncuran *E-kiosk* yaitu mesin penjualan tiket kereta api jarak jauh.

Sebelumnya PT. Kereta Api Indonesia juga melakukan inovasi dengan meluncurkan aplikasi *e-ticketing* yang diberi nama “KAI Access”, sebuah aplikasi *mobile* untuk pemesanan tiket secara online dan mendapatkan info-info terbaru terkait kereta api. Aplikasi ini dapat diunduh di *Google Playstore*, *Appstore*, *Windows Market*, dan *Blackberry App* (KAI, 2016). Peluncuran aplikasi KAI Access tentunya didasarkan pada data jumlah pengguna kereta api di Indonesia yang kian melonjak. Fakta tersebut dikonversi menjadi peluang dan tantangan untuk memuaskan dan memudahkan pengguna transportasi kereta api mendapatkan tiket kereta hanya dengan sekali *klik* saja. Selain meluncurkan aplikasi KAI Access, PT Kereta Api Indonesia melakukan inovasi lain dengan bekerja sama dengan beberapa perusahaan OTA (*Online Travel Agent*) untuk meningkatkan angka penjualan tiket secara online, salah satunya adalah Tiket.com yang mengeluarkan aplikasi yang dapat diunduh di *Google Play Store*, *Appstore*, *Windows Market*, dan *Blackberry App*. Tiket.com telah bekerja sama dengan PT Kereta Api sejak tahun 2012 sebelum aplikasi resmi PT KAI yaitu aplikasi KAI Access diluncurkan, sehingga diharapkan aplikasi KAI Access dapat melakukan perbaikan berdasarkan pencapaian-pencapaian yang didapatkan oleh Tiket.com.

Aplikasi KAI Access dan Tiket.com menawarkan banyak kemudahan, akan tetapi kemudahan-kemudahan tersebut tak lantas membuat pengguna merasa terpuaskan. Terkadang sebuah sistem juga mempunyai kelemahan, untuk itu perlu adanya perbaikan-perbaikan yang perlu dilakukan pada aplikasi KAI Access dengan mengacu pada performa aplikasi Tiket.com. Melalui *review* yang dituliskan oleh pengguna aplikasi diharapkan ada umpan balik dari pengguna, tetapi dengan begitu banyak data *review* yang masuk sistem aplikasi *e-ticketing* belum mampu mengetahui hal-hal apa saja yang perlu dilakukan untuk perbaikan.

Belum ada metode yang sistematis untuk mengklasifikasikan sebuah *review* menjadi kelompok positif dan negatif.

Klasifikasi data *review* pengguna berdasarkan jenis sentimen akan mempermudah pihak PT. KAI dalam mendapatkan informasi persepsi dari pengguna aplikasi KAI Access, kemudian peneliti akan membandingkan sentimen yang didapatkan dengan sentimen aplikasi *e-ticketing* lainnya yaitu Tiket.com. Diharapkan agar aplikasi KAI Access dapat melakukan banyak perbaikan guna meningkatkan kualitas pelayanan terhadap pengguna aplikasi. Proses pengklasifikasian dilakukan dengan *machine learning* metode *Support Vector Machine* (SVM) dan eksplorasi data *review* menggunakan metode statistika deskriptif dan asosiasi teks untuk mencari kata – kata yang berhubungan.

Harapannya dengan penelitian yang dilakukan ini peneliti mampu mengklasifikasikan *review* teks dengan baik sehingga nantinya informasi yang ada dalam kumpulan *review* aplikasi KAI Access dan Tiket.com dapat diekstraksi dengan baik serta memberikan informasi yang berguna bagi pihak yang membutuhkan.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan, maka rumusan masalah dalam penelitian ini adalah sebagai berikut :

1. Bagaimana gambaran umum persepsi pengguna aplikasi KAI Access dan Aplikasi Tiket.com pada situs *google play store*?
2. Bagaimana klasifikasi dan akurasi data *review* pengguna aplikasi KAI Access dan Tiket.com menjadi kelas positif dan negatif dengan menggunakan metode *Support Vector Machine* (SVM)?
3. Informasi apa yang diperoleh dari *review* pengguna aplikasi KAI Access dan Tiket.com berdasarkan hasil klasifikasi yang di dapatkan?
4. Bagaimana rencana pemecahan masalah terhadap aplikasi KAI Access dan Tiket.com berdasarkan diagram sebab akibat?

1.3 Batasan Masalah

Adapun batasan masalah yang digunakan peneliti agar pembahasan dalam penelitian ini tidak menyimpang adalah :

1. Penelitian ini menggunakan data *review* pengguna aplikasi KAI Access dan Tiket.com pada *Google Play Store* pada 1 Januari 2018 – 20 Maret 2018.
2. Penelitian ini mengklasifikasikan *review* aplikasi KAI Access dan Tiket.com berdasarkan kata yang terdapat pada kamus lexicon, tidak berdasarkan frasa atau kalimat.
3. Hasil penelitian ini berupa model yang dapat digunakan pada klasifikasi yang akan mendatang.

1.4 Jenis Penelitian dan Metode Analisis

Penelitian dalam tugas akhir ini termasuk dalam penelitian aplikatif. Metode analisis yang digunakan adalah *Text Mining (Sentiment Analysis)* dengan metode *support vector machine*, asosiasi teks dan diagram sebab akibat untuk mendapatkan rencana pemecahan masalah pada aplikasi KAI Access dan Tiket.com.

1.5 Tujuan Penelitian

Berdasarkan rumusan masalah, dalam penelitian ini ada beberapa tujuan yang ingin dicapai adalah sebagai berikut :

1. Mengetahui gambaran umum persepsi pengguna aplikasi KAI Acces dan Tiket.com pada website *google play store*.
2. Dapat menerapkan analisis sentimen pada data *review* pengguna aplikasi KAI Access dan Tiket.com menjadi kelas positif dan negatif dengan menggunakan metode *Support Vector Machine*.
3. Mendapatkan informasi yang penting dari *review* pengguna aplikasi *e-ticketing* berdasarkan hasil klasifikasi.
4. Mendapatkan rencana pemecahan masalah terhadap aplikasi KAI Access dan Tiket.com berdasarkan diagram sebab akibat.

1.6 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut :

1. Mengetahui gambaran umum tentang tanggapan masyarakat dengan adanya aplikasi *e-ticketing*, terutama pada aplikasi KAI Access dan Tiket.com.
2. Klasifikasi *review* dari pengguna aplikasi dapat mempermudah pihak yang memiliki kepentingan untuk dapat lebih cepat dan efisien mencari apa yang diperlukan.
3. Hasil klasifikasi sentimen dapat dijadikan sebagai bahan evaluasi kepada pihak manajemen PT. KAI (Kereta Api Indonesia) terhadap kepuasan pengguna aplikasi KAI Access yang berguna untuk pengembangan pelayananan perkereta apian di Indonesia.
4. Pihak yang memiliki kepentingan dapat melihat informasi apa yang tersembunyi dari banyaknya kumpulan *review* yang sangat banyak sehingga dapat dilakukan penanganan dan focus kearah evaluasi yang lebih baik.

BAB II

TINJAUAN PUSTAKA

2.1 Transportasi

Transportasi dapat diartikan sebagai usaha yang memindahkan, menggerakkan, mengangkut, atau mengalihkan suatu objek dari satu tempat ke tempat lain, dimana di tempat lain objek tersebut lebih bermanfaat atau dapat berguna untuk tujuan-tujuan tertentu (Miro, 2005). Menurut (Bowersox, 1981) transportasi adalah perpindahan barang atau penumpang dari satu tempat ke tempat lain, dimana produk dipindahkan ke tempat lain, dimana produk dipindahkan ke tempat tujuan dibutuhkan. Menurut (Abbas, 2000) transportasi adalah kegiatan pemindahan barang (muatan) dan penumpang dari suatu tempat ke tempat lain.

2.2 Profil PT Kereta Api Indonesia



Gambar 2. 1 : Logo PT Kereta Api Indonesia

PT Kereta Api Indonesia (Persero) disingkat KAI atau PT KAI adalah Badan Usaha Milik Negara Indonesia yang menyelenggarakan jasa angkutan kereta api. Layanan PT KAI meliputi angkutan penumpang dan barang. (Wikipedia). KAI pada awalnya hanya melaksanakan kegiatan usaha layanan jasa perkeretaapian, namun seiring dengan dinamika dunia usaha dan berkembangnya tuntutan pasar, KAI saat ini juga melaksanakan kegiatan usaha penunjang lainnya dengan memanfaatkan sumber daya yang dimilikinya yaitu meliputi pengelolaan properti yang terkait dengan jasa kereta api, pariwisata berbasis kereta api, restoran di kereta api (*on train services*) dan di stasiun, termasuk jasa catering dan distribusi logistik. Dalam menjalankan bisnisnya, KAI terus berupaya menerapkan standar terbaik di bidangnya berdasarkan sistem manajemen yang berlaku (KAI, 2016).

2.3 *E-ticketing*

Electronic ticketing atau *e-ticketing* adalah suatu cara untuk mendokumentasikan proses penjualan dari aktifitas perjalanan pelanggan tanpa harus mengeluarkan dokumen berharga secara fisik ataupun *paper ticket* (Ng-Kruelle, Swatman, & Kruelle, 2006) sedangkan e-ticket atau electronic ticket itu sendiri adalah tiket yang ditampilkan dalam catatan digital tanpa harus tercetak dengan kertas. Penggunaan *e-ticket* dapat memperkecil biaya pencetakan tiket yang dilakukan oleh perusahaan transportasi.

2.3.1 KAI Access

KAI Access merupakan aplikasi *mobile* resmi yang dikembangkan oleh PT Kereta Api Indonesia. Aplikasi ini menawarkan berbagai kemudahan bagi penggunaanya dalam mengakses informasi seputar kereta api serta pemesanan tiket secara online (Mudikgratis.com, 2017). Aplikasi ini sendiri mulai diluncurkan sejak 4 September 2014 dan dapat diunduh di Google *Playstore*, *Appstore*, *Windows Market*, dan *Blackberry App* (KAI, 2016).



Gambar 2. 2 : Logo Aplikasi KAI Access

Fitur utama yang ditawarkan KAI Access adalah pemesanan tiket secara online, pemesan tiket dapat dilakukan dimana saja dan kapan saja saat terhubung internet. Fitur ini menjawab kebutuhan masyarakat yang ingin mendapatkan tiket kereta api secara lebih mudah dan manusiawi tanpa harus antri berdesakan bahkan menginap di loket stasiun. Pengguna hanya perlu memasukkan kota asal dan kota tujuan untuk mencari tiket kereta api, kemudian akan ditampilkan semua kereta dan jumlah tiket yang masih tersedia juga akan dicantumkan.

Selain memesan tiket secara online, pengguna aplikasi KAI Access juga bisa langsung mendapatkan tiket online atau biasa disebut *e-ticket*. Fungsi *e-ticket*

sama halnya dengan tiket kertas yang biasa di dapatkan. Kode *booking* yang tertera pada *e-ticket* dapat digunakan untuk mencetak tiket sendiri melalui *booth CTM* (Cetak Tiket Mandiri) yang terdapat di stasiun atau bisa langsung mencetak *boarding pass ticket*, tetapi fitur terbaru dari aplikasi KAI Access tidak mengharuskan mencetak *boarding pass*. Kini tersedia *e-boarding pass* yang memudahkan pengguna transportasi moda kereta api. Penumpang cukup menunjukkan *e-boarding pass* yang terpampang pada layar *smartphone* mereka kepada petugas *boarding*, tanpa harus ke mesin cetak *boarding pass* CIC (*Check In Counter*).

E-boarding pass hanya dapat diterbitkan khusus pemesanan tiket kereta api via KAI Access versi terbaru dan dapat diunduh mulai dari dua jam sebelum keberangkatan kereta api. Jika sudah menerbitkan *e-boarding pass* melalui aplikasi KAI Access, otomatis penumpang tidak dapat mencetak *boarding pass* di mesin CIC di stasiun, atau sebaliknya (Suprpto, 2017).

Fitur yang ada pada aplikasi KAI Access tak hanya sekedar pemesanan tiket dan pembelian tiket secara *online*, tetapi ada sebuah fitur *notification* yang berguna untuk mengingatkan penumpang agar tidak terlewat di stasiun tujuan. Aplikasi KAI Access tak hanya ingin memudahkan pengguna dalam hal tiket kereta tetapi dalam hal pelayanan dari mulai masuk kedalam kereta sampai keluar dari kereta akan dilayani dengan sungguh-sungguh oleh pihak PT Kereta Api Indonesia sebagai pengembang aplikasi KAI Acces. Aplikasi KAI Acces pada android tersedia di *Google Play Store*, pembaruan yang dilakukan pihak PT.KAI sebagai upaya peningkatan kinerja aplikasi sudah dilakukan beberapa kali sejak pertama kali rilis.

2.3.2 Tiket.com



Gambar 2. 3 : Logo Aplikasi Tiket.com

Tiket.com adalah salah satu perusahaan OTA (*Online Travel Agent*) yang menyediakan dan menyelenggarakan sistem dan fasilitas pemesanan dan pembelian tiket secara *online*. (Tiket.com, 2011). Tiket.com didirikan oleh Wenas Agusetiawan, Dimas Surya Yaputra, Natali Ardianto, Mikhael Gaery Undarsa pada bulan Agustus 2011. Pada awalnya, Wenas Agusetiawan membeli domain tiket.com dan mengajak beberapa kolega untuk membuat sebuah situs web untuk melayani pemesanan tiket pesawat dan kereta api. Tiket.com merupakan salah satu rekan bisnis Kereta Api Indonesia di Indonesia. Tiket.com juga bekerjasama dengan berbagai maskapai di Indonesia untuk memudahkan orang untuk pemesanan tiket pesawat melalui internet.

2.4 Google Play



Gambar 2. 4 : Logo Google Play

Google Play adalah layanan konten digital toko aplikasi online milik Google yang menawarkan produk-produk seperti musik, buku, dan aplikasi. Layanan Google Play dapat diakses melalui tiga cara yaitu melalui web, aplikasi android (Play Store), dan Google TV . Google Play pertama dirilis pada tanggal 22 Oktober 2008, awalnya bernama Android Market. Penggantian Android Market dengan Google Play dilakukan pada bulan Maret 2012. Android Market akan disebut Play Store, sedangkan konten buku, musik serta film disebut Play Books, Play Music dan Play Movies (Wikipedia). Hal yang tidak kalah penting pada google play adalah adanya fitur ulasan yang berisi *review* dari para pengguna.

2.5 Penelitian Terdahulu

Penelitian (Saraswati, 2011) yang berjudul “Text Mining dengan Metode Naive Bayes Classifier dan Support Vector machines untuk Sentiment Analysis” mengklasifikasikan opini pada kelas positif dan negatif pada data berbahasa

inggris dan bahasa Indonesia menggunakan metode Naïve Bayes Classifier dan Support Vector Machine. Hasil percobaan klasifikasi data menunjukkan bahwa metode SVM memberikan unjuk kerja yang lebih baik dari pada metode NBC dalam sentimen analysis.

Penelitian yang dilakukan oleh (Hidayatullah, 2014) ini melakukan analisis sentimen dan klasifikasi tweet berbahasa Indonesia terhadap tokoh publik menggunakan metode Naive Bayes Classifier. Klasifikasi tweet pada penelitian ini dilakukan berdasarkan sentimen terhadap fitur tokoh publik. Metode Support Vector Machine juga menghasilkan akurasi performansi yang lebih baik daripada metode Naive Bayes. Namun demikian, secara keseluruhan penggunaan metode Support Vector Machine dan Naive Bayes sama-sama memiliki performansi yang cukup baik untuk melakukan klasifikasi tweet.

(Ramadhani, 2015) menggunakan metode probabilistic Naïve Bayes Classifier dan metode Support Vector Machine pada skripsinya yang berjudul “Analisis Sentimen menggunakan Metode Naïve Bayes Classifier dengan Model Dokumen Bernoulli dan Support Vector Machine” untuk menentukan kelas suatu dokumen secara biner yaitu kelas positif dan kelas negatif. Data yang telah diklasifikasi kemudian dihitung tingkat akurasi kebenarannya menggunakan metode support vector machine dengan menentukan fungsi Kernel serta proporsi untuk data training dan data testing yang sesuai. Dari perbandingan nilai akurasi klasifikasi dengan menggunakan kedua metode diatas, didapatkan nilai akurasi paling tinggi adalah dengan menggunakan metode naïve bayes classifier (NBC) dengan akurasi sebesar 62,6295%. Dari 5.412 dokumen training dan 2701 dokumen testing yang telah diklasifikasi, didapatkan proporsi opini untuk kelas positif sebesar 44,46501% dan opini untuk kelas negatif sebesar 55,53499%.

Penelitian yang dilakukan oleh (Olabenjo, 2016) membuat dua variasi pengelompokan Naïve Bayes menggunakan data dari Google Play Store dan dari tempat lain untuk mengklasifikasikan aplikasi baru. Pengelompokan ini kemudian dievaluasi dengan beberapa metode evaluasi dan hasilnya dibandingkan satu dengan yang lain. Hasilnya menunjukkan bahwa metode Naïve Bayes baik untuk

masalah klasifikasi aplikasi di Google Play Store dan mengotomatisasi pengkategorian aplikasi android di Google Play Store.

Klasifikasi metode Support Vector Machine pernah digunakan pada penelitian data LAPOR! yang dilakukan oleh (Ulwan, 2016), hasil klasifikasi menunjukkan tingkat akurasi sebesar 96.7%. Data diklasifikasikan menjadi tiga yaitu pada kelas aspirasi, keluhan, dan pertanyaan.

Analisis sentimen tentang twitter pernah dilakukan oleh (D.Suganthi & Dr.A.Geetha, 2017) yang berkaitan dengan The Goods and Service Tax (GST) tentang sistem pajak di India. Tweet diklasifikasikan menjadi tweet positif, negatif, dan netral berdasarkan sentimen dengan mengklasifikasikan dataset dengan menggunakan berbagai algoritma Support Vector Machine (SVM). Analisis sentimen di R dengan menghitung skor sentimen untuk masing-masing tweet.

Penelitian dengan menggunakan metode Support Vector Machine untuk klasifikasi sentimen pernah dilakukan oleh (Abtohi, 2017) pada data ulasan hotel pada situs Tripadvisor. Hasil klasifikasi ulasan dengan metode SVM menunjukkan tingkat akurasi sebesar 95,27% pada ulasan berbahasa Inggris dan sebesar 95,00% untuk ulasan berbahasa Indonesia.

Review pada Google Play Store sebelumnya juga pernah diteliti oleh (Fanani, 2017). Data *review* diklasifikasikan menjadi SPAM dan bukan SPAM. *Review* SPAM dapat merugikan para pengguna dan mengganggu para developer yang menggunakan *review* tersebut sebagai bahan dalam melakukan evaluasi terhadap aplikasinya. Tahap yang dilakukan peneliti adalah preprocessing yang akan mengolah data mentah menjadi data yang siap untuk diolah dan data tersebut diubah menjadi matriks lalu dilakukan pembuatan vector. Algoritma yang digunakan pada penelitian ini adalah Support Vector Machine (SVM) dan Naïve Bayes Classifier (NBC). Hasil penelitian menunjukkan metode Support Vector Machine menghasilkan akurasi lebih baik dibandingkan dengan metode Naïve Bayes Classifier.

Tabel 2.1 merupakan tabel rangkuman perbandingan dengan penelitian sebelumnya yang berkaitan text mining khususnya metode support vector machine.

Tabel 2. 1 : Perbandingan Penelitian Terdahulu

No	Penulis	Judul	Penelitian
1.	(Saraswati, 2011)	Text Mining dengan Metode Naive Bayes Classifier dan Support Vector machines untuk Sentiment Analysis	Penggunaan Naïve Bayes Classifier dan Support Vector Machine untuk analisis sentiment.
2.	(Hidayatullah, 2014)	Analisis Sentimen dan Klasifikasi Kategori Terhadap Tokoh Publik pada Data Twitter Menggunakan Naive Bayes Classifier	Penggunaan Naïve Bayes Classifier untuk analisis klasifikasi sentiment terhadap tokoh public pada twitter
3.	(Ramadhani, 2015)	Analisis Sentimen menggunakan Metode Naïve Bayes Classifier dengan Model Dokumen Bernoulli dan Support Vector Machine	Penggunaan Analisis Sentimen pada dokumen teks menggunakan metode NBC
4.	(Olabenjo, 2016)	Applying Naive Bayes Classification to Google Play Apss categorization	Penggunaan Naïve Bayes Classifier untuk mengkategorikan aplikasi pada Google Play
5.	(Abtohi, 2017)	Implementasi Teknik Web Scrapping dan Klasifikasi Sentiment Menggunakan Metode Support Vector Machine dan Asosiasi	Penggunaan SVM untuk klasifikasi sentiment pengguna hotel.
6.	(Fanani, 2017)	Klasifikasi <i>Review</i> Software pada Google Play Store Menggunakan Pendekatan Analisis Sentimen	Penggunaan Support Vector Machine dan Naïve Bayes Classifier untuk mengklasifikasikan <i>review</i> software menjadi SPAM dan bukan SPAM.

No	Penulis	Judul	Penelitian
7.	(D.Suganthi & Dr.A.Geetha, 2017)	Twitter Sentiment Analisis on GST tweets using R tool	Penggunaan Support Vector Machine untuk mengklasifikasikan sentimen pada data twitter.
8.	(Ulwan, 2016)	Pattern Recognition pada Unstructured Data Teks Menggunakan Support Vector Machine dan Association	Penggunaan SVM untuk klasifikasi data LAPOR! menjadi tiga kelas klasifikasi.

BAB III

LANDASAN TEORI

3.1 *Online Review*

Dalam Kamus Besar Bahasa Indonesia, *review* berarti ulasan memiliki pengertian kupasan, tafsiran, komentar, sedangkan kata *online* memiliki arti *controlled by or connected to a computer* sesuai dengan pengertian yang dijelaskan dalam *Oxford Dictionaries online* yang apabila diterjemahkan yaitu dikontrol oleh atau terkoneksi pada sebuah komputer.

Review mempunyai arti sebagai tinjauan atau ringkasan dari beberapa sumber baik buku, film, berita, dan lain-lain. *Review* difungsikan sebagai salah satu alat untuk meninjau kualitas suatu karya. (Pardiyono, 2007) menyatakan bahwa teks ulasan adalah teks yang berisi pemberian kritik, evaluasi, atau melakukan ulasan terhadap karya cipta intelektual. Teks ini bertujuan untuk memberikan kritikan, hasil evaluasi, atas suatu karya ilmiah, buku, atau karya seni. Teks *review* adalah tulisan yang isinya menimbang atau menilai sebuah karya yang dikarang atau dicipta orang lain.

Dengan demikian *online review* dapat diartikan sebagai tinjauan atau ringkasan yang berisi kritik, saran atau dukungan pada suatu sumber yang terkoneksi pada sebuah jaringan komputer yang bersifat *online*.

3.2 Analisis Deskriptif

Analisis deskriptif adalah cara penggambaran suatu masalah berdasarkan kumpulan data dengan cara menyusun data tersebut sedemikian rupa sehingga dapat dipahami dengan mudah karakteristik datanya yang berguna untuk keperluan selanjutnya, dengan kata lain terdapat suatu proses pengumpulan data dan pengolahan data berdasarkan tujuan. Pengolahan data pada analisis deskriptif hanya berhubungan dengan menguraikan atau memberikan keterangan pada suatu data secara umum atau menggambarkan secara umum dari data yang didapatkan.

Menurut (Santosa, 2004) menjelaskan bahwa statistika deskriptif adalah ilmu yang menggunakan metode numerik dan grafik untuk mencari pola data, meringkas informasi yang ada pada kumpulan data serta menyampaikan informasi

dalam bentuk sesuai dengan apa yang diinginkan. Statistika deskriptif adalah metode-metode yang berkaitan dengan pengumpulan data dan penyajian data sehingga memberikan informasi yang berguna (Walpole & Myers, 1995). Analisis deskriptif adalah statistik yang digunakan menganalisis data dengan cara mendeskripsikan data atau menggambarkan data yang telah terkumpul tanpa membuat kesimpulan yang berlaku secara umum (Sugiyono, 2004).

3.3 Data Mining dan Klasifikasi

Data mining adalah kegiatan mengekstraksi pengetahuan dari data yang berjumlah besar, informasi inilah yang nanti sangat berguna untuk pengembangan (Fadli, 2011). Data mining juga dengan istilah penambangan data, ekstraksi pola data atau pengerukan data untuk memperoleh pengetahuan dari data (Han, Kamber, & Pei, 2012). Selain itu data mining diartikan sebagai proses menemukan pola wawasan yang menarik dan baru sebagai model deskriptif yang dapat dipahami dan di prediksi dari data dengan skala yang besar (Zaki & JR, 2014). Data mining disebut sebagai proses *knowledge discovery in database* (KDD) pada gambar 3.1.

Proses KDD secara rinci ditunjukkan dengan langkah-langkah berikut (Han, Kamber, & Pei, 2012) :

a. *Data cleaning*

Pembersihan data untuk menghilangkan data noise yang mengganggu dan data yang tidak konsisten.

b. *Data Integration*

Integrasi data merupakan proses mengkombinasi beberapa sumber data. Dilakukan penggabungan data dari sumber data yang masih terpecah untuk dibentuk penyimpanan data yang kohonen.

c. *Data selection*

Seleksi data merupakan proses pengambilan data yang sesuai dengan tugas analisis dari basis data. Pada tahap seleksi dilakukan pengurangan representasi dari data dan meminimalkan hilangnya informasi data yang akan dianalisis melalui pengurangan atribut dan kompresi data.

d. *Data transformation*

Transformasi data adalah suatu proses dimana data berubah atau bersatu menjadi bentuk yang tepat untuk menambang (*mining*) dengan ringkasan performa.

e. *Data mining*

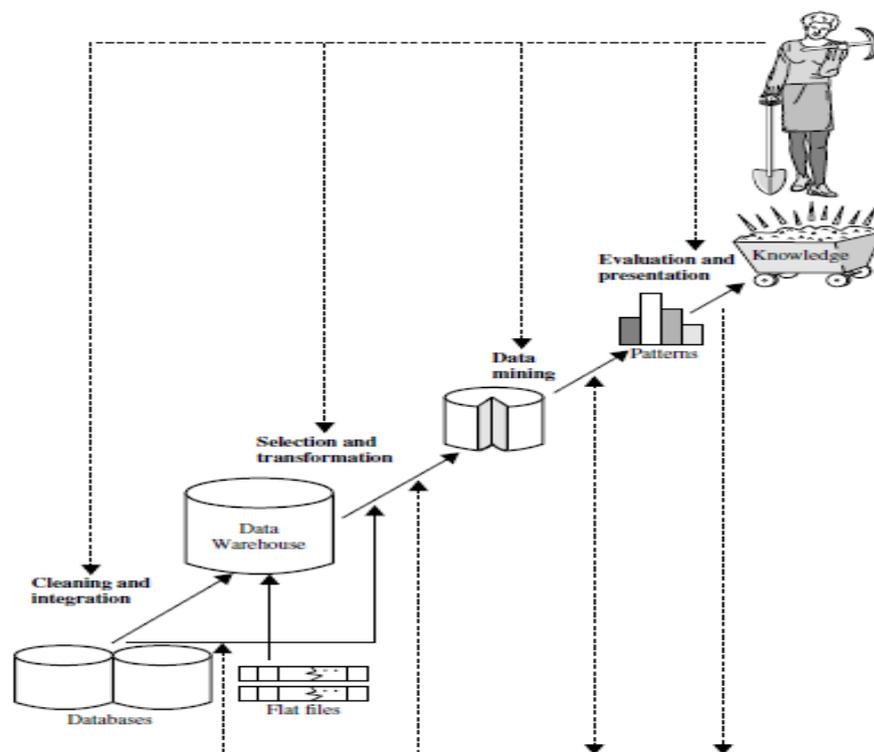
Penambangan data adalah proses mencari pola data yang menarik dalam data yang telah dipilih menggunakan suatu teknik dan algoritma tertentu. Pemilihan algoritma ini bergantung pada tujuan dan proses yang akan dituju.

f. *Pattern evaluation*

Evaluasi pola adalah proses identifikasi pola menarik yang mewakili pengetahuan berdasarkan beberapa tindakan yang menarik.

g. *Knowledge presentation*

Representasi pengetahuan adalah proses presentasi gambaran teknik visual dan pengetahuan kepada pengguna.



Sumber : (Han, Kamber, & Pei, 2012)

Gambar 3. 1 : Tahap-Tahap Knowledge Discovery from Data (KDD)

Machine Learning adalah ilmu yang mempelajari bagaimana komputer dapat belajar atau meningkatkan kinerja berdasarkan data agar secara otomatis mengenali pola data untuk membuat keputusan cerdas berdasarkan data (Han, Kamber, & Pei, 2012). *Machine learning* adalah suatu area dalam *artificial intelligence* yang berhubungan dengan pengembangan teknik-teknik yang bisa diprogramkan dan belajar dari data masa lalu. Pengenalan pola, data mining, dan machine learning sering dipakai untuk menyebut sesuatu yang sama atau bersinggungan dengan ilmu probabilitas dan statistik kadang juga optimasi. *Machine learning* sebagai alat analisis dalam data mining (Santoso, 2007).

Machine learning adalah disiplin ilmu yang tumbuh cepat, adapun beberapa algoritma dalam *machine learning* berdasarkan masukan dan keluarannya menurut (Han, Kamber, & Pei, 2012) adalah sebagai berikut :

- a. *Supervised learning*
- b. *Unsupervised learning*
- c. *Semi-supervised learning*
- d. *Active learning*

Salah satu metode data mining yang termasuk dalam *supervised learning* adalah klasifikasi karena prosesnya melalui pembelajaran data masa lalu. Klasifikasi digunakan untuk memprediksi kategori dari sekumpulan data dengan berbagai atribut. Tujuan klasifikasi adalah membangun model yang menguraikan dan membedakan kelas untuk peramalan kedepan (Fanani, 2017).

3.4 Analisis Sentimen

Analisis sentimen atau *opinion mining* adalah bidang ilmu data mining yang mempunyai tujuan untuk menganalisis, memahami, mengolah dan mengekstrak data tekstual yang berupa opini, sentimen, evaluasi, sikap, dan emosi terhadap suatu entitas seperti produk, servis, organisasi, individu, dan topik tertentu. Klasifikasi sentimen digunakan untuk menyelesaikan masalah klasifikasi dua kelas, positif dan negatif. Data pengujian yang digunakan biasanya adalah ulasan produk secara *online*. Karena ulasan *online* memiliki nilai penilaian yang ditetapkan oleh para peneliti. Sebagian besar makalah penelitian tidak

menggunakan kelas netral, dengan tujuan agar memudahkan masalah klasifikasi, namun juga tidak jarang peneliti yang menggunakan kelas netral, misalnya penggunaan peringkat bintang tiga sebagai kelas netral (Liu, 2012).

3.5 Text Mining

Text mining adalah teknologi yang mampu menganalisa data teks semi-terstruktur maupun tidak terstruktur, sedangkan data mining mengolah data yang terstruktur (Han & Kamber, 2012).

Teks yang dilakukan pada proses *text mining*, pada umumnya memiliki beberapa karakteristik berdimensi tinggi, terdapat *noise* pada data, dan terdapat struktur teks yang tidak baik. Untuk mendapatkan bentuk data yang siap diproses oleh data mining, maka harus dilakukan *text preprocessing* terlebih dahulu dengan cara penyeleksian kata. Setiap kata dipecah menjadi bagian yang lebih kecil sehingga mempunyai arti yang lebih sempit.

3.5.1 Spelling Normalization

Proses perbaikan kata-kata yang masih salah dalam pengejaan atau merupakan singkatan dalam bentuk tertentu. Perbaikan kata dilakukan agar jumlah perhitungan dimensi kata tidak melebar. Perhitungan jumlah dimensi kata akan melebar jika kata yang salah eja atau kata singkatan tidak dirubah karena akan dihitung sebagai kata yang berbeda pada proses penyusunan matriks.

3.5.2 Case Folding

Case folding adalah mengubah semua huruf dalam dokumen menjadi satu bentuk yang sama, misalnya dokumen diubah menjadi huruf kecil atau sebaliknya dan hanya huruf 'a' sampai huruf 'z' yang diterima. Selain karakter huruf dihilangkan.

3.5.3 Tokenizing

Proses pemisahan teks menjadi potongan kalimat dan kata yang disebut token. Tujuan dari proses ini adalah mendapatkan potongan kata yang akan menjadi entitas yang memiliki nilai dalam matriks dokumen teks yang akan dianalisis.

3.5.4 *Filtering*

Proses penghilangan kata-kata serta tanda-tanda yang tidak bermakna secara signifikan, seperti *hashtag* (#), url, tanda baca tertentu (*emoticon*), dan lainnya (Megawati, 2015).

3.6 Term Frequency-Inverse Document Frequency (TF-IDF)

Setelah data melalui tahap *preprocessing*, data perlu dibuat model agar data yang masih berupa kata-kata dapat diolah dan dihitung. Data kata-kata tersebut diubah menjadi vektor kemudian diberi nilai dan pembobotan untuk setiap kata untuk kemudian diolah menggunakan algoritma prediksi.

Term Frequency (TF) adalah frekuensi kemunculan sebuah term dalam sebuah dokumen. Semakin besar jumlah term yang muncul (TF tinggi) maka semakin besar bobot dokumen atau memberikan nilai kesesuaian yang semakin besar (Informatikalogi, 2016). Terdapat beberapa jenis formula *Term Frequency* (TF) :

a. TF biner (*binary TF*)

Hanya memperhatikan ada tidaknya kata atau term pada sebuah dokumen, jika ada diberi nilai satu (1) jika tidak ada diberi nilai nol (0).

b. TF murni (*raw TF*)

Nilai TF didasarkan pada jumlah kemunculan term atau kata dalam dokumen. Contohnya jika muncul tiga (3) kali maka kata tersebut bernilai tiga (3).

c. TF logaritmik

Perhitungan nilai TF ini untuk menghindari dominasi dokumen yang mengandung sedikit term dalam query tetapi mempunyai frekuensi yang tinggi.

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

d. TF normalisasi

Perhitungan ini menggunakan perbandingan antara frekuensi kata dengan jumlah semua kata pada sebuah dokumen.

$$tf = 0.5 + 0.5x\left(\frac{tf}{maxtf}\right) \quad (2)$$

Inverse Document Frequency (IDF) merupakan metode statistik numeric yang menghitung seberapa pentingnya kata dalam sebuah dokumen. Metode ini digunakan sebagai bobot dalam pencarian informasi dalam text mining. (Fanani, 2017). Formula IDF dapat dituliskan sebagai berikut :

$$idf_j = \log\left(\frac{D}{df_j}\right) \quad (3)$$

dimana :

D : Jumlah semua dokumen dalam koleksi

df_i : Jumlah dokumen yang mengandung term t_j

Didapatkan rumus umum TF-IDF dengan menggabungkan perhitungan raw TF dan formula IDF dengan mengalikan keduanya :

$$w_{ij} = tf_{ij} \times idf_j \quad (4)$$

$$w_{ij} = tf_{ij} \times \log\left(\frac{D}{df_j}\right) \quad (5)$$

w_{ij} : Bobot term t_j terhadap dokumen d_i

tf_{ij} : Jumlah kemunculan term t_j dalam dokumen d_i

D : Jumlah semua dokumen dalam koleksi

df_i : Jumlah dokumen yang mengandung term t_j

(minimal ada stu kata yaitu term t_j)

Perhitungan (TF-IDF) akan dijelaskan berdasarkan rumus yang telah dijabarkan sebelumnya, pembobotan dilakukan dengan menggunakan tiga (3) dokumen berdasarkan kata ‘mudah’, ’murah’, dan ‘harga’ sebagai berikut :

D_1 : Aplikasi ini mudah untuk mencari tiket kereta dengan harga murah

D_2 : Harga tiket kereta di Tiket.com murah

D_3 : Aplikasi yang mudah dan harga tiket murah

Ketiga dokumen tersebut masuk dalam tahap preprocessing sehingga akan menghasilkan dokumen baru dengan perubahan sebagai berikut :

D_1 : aplikasi mudah mencari tiket kereta harga murah

D_2 : harga tiket kereta tiket.com murah

D_3 : aplikasi mudah harga tiket murah

Sehingga terbentuk beberapa term dari hasil *preprocessing* ketiga dokumen yang dijadikan sebagai contoh adalah aplikasi, mudah, mencari, tiket, kereta, harga, murah, dan tiket.com. Tahap selanjutnya adalah mewujudkan dokumen dalam bentuk vektor dengan elemen sebanyak *term query* yang ada pada setiap dokumen. Vektor tersebut merupakan bobot dari setiap query yang dihitung berdasarkan metode TF-IDF. Contoh perhitungan TF-IDF dari term ‘mudah’ pada dokumen satu (D_1) adalah sebagai berikut :

$tf_{mudah} = 1$ Jumlah kemunculan term ‘mudah’ dalam dokumen d_1

$D = 3$ Jumlah semua dokumen dalam koleksi

$df_{mudah} = 2$ Jumlah dokumen yang mengandung term ‘mudah’

Perhitungan diawali dengan mencari nilai IDF dari term ‘mudah’ dengan persamaan :

$$idf_{mudah} = \log\left(\frac{D}{df_{mudah}}\right) = \log\left(\frac{3}{2}\right) = 0.176$$

Sehingga diperoleh nilai IDF pada term ‘mudah’ adalah sebesar 0.176, kemudian akan dihitung nilai TF-IDF dari term ‘mudah’ dengan persamaan :

$$tfidf_{mudah} = tf_{mudah} \times idf_{mudah}$$

$$tfidf_{mudah} = 1 \times 0.176 = 0.176$$

Perhitungan TF-IDF dihitung pada tiap kata pada tiap dokumen kemudian dijumlahkan pada setiap dokumen. Sehingga bobot pada dokumen satu (1) adalah sebagai berikut :

$$w_{d1} = tfidf_{mudah} + tfidf_{murah} + tfidf_{harga}$$

$$w_{d1} = 0.176 + 0 + 0.477 = 0.653$$

Selanjutnya dapat dihitung nilai bobot untuk setiap *term* pada *query* dalam masing-masing dokumen adalah seperti tabel 3.1:

Tabel 3. 1 : Perhitungan TF-IDF

Term	TF			df	IDF	TF-IDF		
	D ₁	D ₂	D ₃			D ₁	D ₂	D ₃
mudah	1	0	1	2	0.176	0.176	0	0.176
murah	1	1	1	3	0	0	0	0
harga	1	0	0	1	0.477	0.477	0	0
Nilai bobot masing - masing dokumen						0.653	0	0.176

3.7 Algoritma Prediksi Support Vector Machine

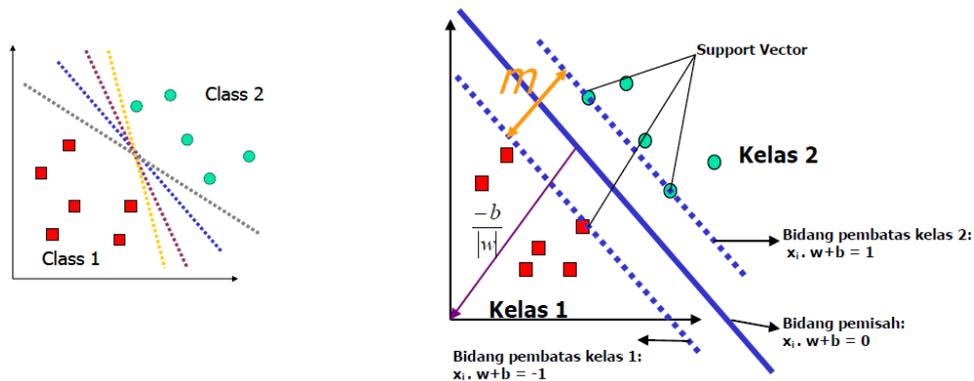
Setelah melakukan *preprocessing* dan vektorisasi data text, yang harus dilakukan dalam mengklasifikasikan dokumen-dokumen tersebut adalah menggunakan sebuah algoritma. Algoritma yang akan digunakan pada penelitian adalah *Support Vector Machine* (SVM). Menurut (Christianini & John S, 2000) *support vector machine* adalah sebuah sistem pembelajaran yang menggunakan ruang hipotesis berupa fungsi-fungsi linear dalam ruang fitur (*fitur space*) berdimensi tinggi, dilatih dengan algoritma pembelajaran berdasarkan pada teori optimasi dengan mengimplementasikan learning bias yang berasal dari teori pembelajaran statistik.

SVM adalah salah satu metode *machine learning* yang bekerja pada prinsip *Structural Risk Minimization* (SRM) dengan tujuan untuk menemukan *hyperlane* terbaik memisahkan dua kelas pada input space. SVM dapat digunakan untuk klasifikasi yang diterapkan pada deteksi tulisan tangan, pengenalan objek, identitas suara dan lain-lain (Ulwan, 2016).

3.7.1 SVM Linearly Separable Data

Linearly separable data adalah data yang dapat dipisahkan secara linear. Misalkan $\{x_1, \dots, x_n\}$ adalah dataset dan $y_i \in \{+1, -1\}$ adalah label kelas dari data x_i . Pada gambar 3.2 menunjukkan berbagai alternatif bidang pemisah yang dapat memisahkan semua data set sesuai dengan kelasnya, namun bidang pemisah

terbaik adalah bidang yang memisahkan data dan memiliki margin yang besar (Sembiring, 2007).



Sumber : (Sembiring, 2007)

Gambar 3. 2 : Alternatif Bidang Pemisah dan Bidang Pemisah Terbaik

Data yang berada pada bidang pembatas disebut *support vector*. Pada visualisasi terdapat dua kelas yang dapat dipisahkan oleh sepasang bidang pembatas yang sejajar. Bidang pembatas pertama membatasi kelas pertama sedangkan pada pembatas bidang kedua membatasi kelas kedua sehingga diperoleh persamaan :

$$\begin{aligned} x_i \cdot w + b &\geq +1, \text{ untuk } y_i = +1 \\ x_i \cdot w + b &\leq -1, \text{ untuk } y_i = -1 \end{aligned} \quad (6)$$

Dimana w adalah bidang normal dan b adalah posisi bidang relatif terhadap pusat koordinat. Nilai margin atau jarak antara bidang pembatas berdasarkan rumus jarak garis ke titik pusat adalah $\frac{1-b-(-1-b)}{w} = \frac{2}{|w|}$. Nilai margin telah dimaksimalkan dengan tetap memenuhi (6). Mengalikan b dan w dengan konstanta sehingga dihasilkan nilai margin yang dikalikan dengan konstanta yang sama. Konstrain (6) merupakan *scaling constraint* yang dipenuhi dengan *rescaling* b dan, selain itu karena memaksimalkan $\frac{1}{|w|}$ sama dengan meminimumkan $|w|^2$ dan jika kedua bidang pembatas pada direpresentasikan dalam pertidaksamaan (7),

$$y_i(x_i \cdot w + b) - 1 \geq 0 \quad (7)$$

sehingga pencarian bidang pemisah terbaik dengan nilai margin terbesar dapat dirumuskan menjadi optimasi konstrain,

$$\begin{aligned} \min \frac{1}{2} |w|^2 \\ \text{s.t. } y_i(x_i \cdot w + b) - 1 \geq 0 \end{aligned} \quad (8)$$

Agar lebih mudah diselesaikan persoalan ini dapat diselesaikan dengan formula *lagrangian* dengan menggunakan *langrange multiplier*, sehingga dapat dirubah menjadi :

$$\min_{w,b} L_p(w, b, \alpha) \equiv \frac{1}{2} |w|^2 - \sum_{i=1}^n \alpha_i y_i (x_i \cdot w + b) + \sum_{i=1}^n \alpha_i \quad (9)$$

$\alpha_i \geq 0$ merupakan nilai koefisien *lagrange*. Kemudian dengan meminimumkan L_p terhadap w dan b , maka $\frac{\partial}{\partial b} L_p(w, b, \alpha) = 0$ diperoleh (10) dan dari $\frac{\partial}{\partial w} L_p(w, b, \alpha) = 0$ diperoleh (11).

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad (10)$$

$$w = \sum_{i=1}^n \alpha_i y_i x_i \quad (11)$$

Vektor w sering bernilai besar bahkan tak terhingga, tetapi nilai α_i terhingga, sehingga formula *lagrangian* L_p (*primal problem*) diubah kedalam dual problem L_D . Dengan mensubsitusikan (11) persamaan ke L_p diperoleh dual problem L_D dengan konstrain yang berbeda.

$$L_D(\alpha) \equiv \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^n \alpha_i \alpha_j y_i y_j x_i x_j \quad (12)$$

$\min_{w,b} L_p = \max_{\alpha} L_D$ jadi persoalan untuk pencarian pemisahan bidang terbaik dirumuskan sebagai berikut:

$$\max_{\alpha} L_D \equiv \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^n \alpha_i \alpha_j y_i y_j x_i x_j \quad (13)$$

$$s. t \sum_{i=1}^n \alpha_i y_i = 0, \alpha_i \geq 0 \quad (14)$$

Diperoleh nilai α_i yang nantinya digunakan untuk menemukan w . Terdapat nilai α_i untuk setiap data latih. Data latih yang memiliki nilai $\alpha_i > 0$ adalah merupakan *support vector* sedangkan sisanya memiliki nilai $\alpha_i = 0$, dengan demikian fungsi keputusan yang dihasilkan hanya dipengaruhi oleh *support vector*.

Formula untuk pencarian bidang pemisah terbaik adalah permasalahan *quadratic programming* sehingga nilai maksimum global dari α_i selalu bisa ditemukan. Setelah nilai α_i ditemukan, maka kelas dari pengujian x dapat ditentukan berdasarkan nilai fungsi keputusan :

$$f(x_d) = \sum_{i=1}^{ns} \alpha_i y_i x_i \cdot x_d + b \quad (15)$$

dimana :

- $\alpha_i =$ Koefisien *lagrange*
- $x_i =$ Support vector
- $y_i =$ Kelas data
- $ns =$ Jumlah Support vector
- $x_d =$ Data yang diklasifikasikan

3.7.2 SVM Nonlinearly Separable Data

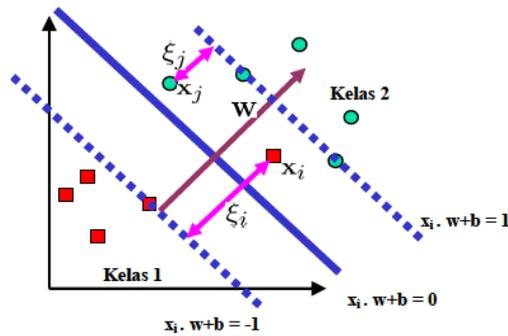
Formula SVM harus dimodifikasi agar solusi dapat ditemukan pada kasus klasifikasi data yang tidak dapat dipisahkan secara linear. Sehingga pada kedua bidang pembatas harus diubah agar lebih fleksibel dengan penambahan variabel ξ_i ($\xi_i \geq 0, \forall i; \xi_i = 0$ jika x_i diklasifikasikan dengan benar) menjadi

$$\begin{aligned} x_i \cdot w + b &\geq 1 - \xi_i, \text{ untuk kelas 1} \\ x_i \cdot w + b &\leq -1 + \xi_i, \text{ untuk kelas 2} \end{aligned} \quad (16)$$

Bidang pemisah terbaik dicari dengan menambahkan variabel ξ_i disebut dengan *soft margin hyperlane*. Sehingga formula berubah menjadi :

$$\begin{aligned} \min & \frac{1}{2} |w|^2 + C \left(\sum_{i=1}^n \xi_i \right) \\ \text{s. t. } & y_i(x_i \cdot w + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \end{aligned} \quad (17)$$

C adalah parameter yang menentukan besarnya penalti kesalahan dalam klasifikasi data. Bentuk persoalan (17) memenuhi prinsip SRM dimana meminimumkan $\frac{1}{2} |w|^2$ ekuivalen dengan meminimumkan dimensi VC dan meminimumkan $C(\sum_{i=1}^n \xi_i)$ berarti meminimumkan *error* pada data latih.



Sumber : (Sembiring, 2007)

Gambar 3. 3 : Soft Margin Hyperplane

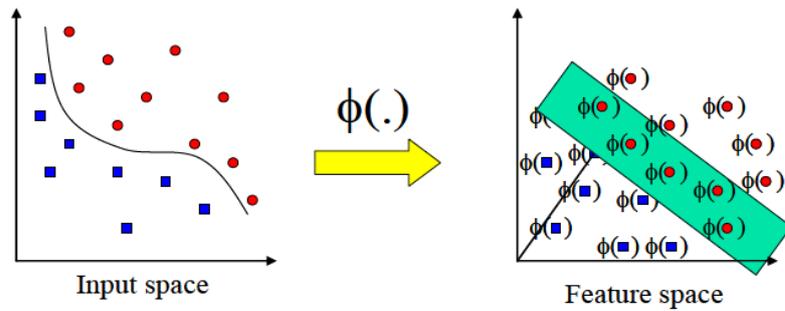
Bentuk *primal problem* sebelumnya berubah menjadi :

$$\min_{w,b} L_p(w, b, \alpha) \equiv \frac{1}{2} |w|^2 + C \left(\sum_{i=1}^n \xi_i \right) - \sum_{i=1}^n \alpha_i \{y_i(x_i \cdot w + b) - 1 + \xi_i\} - \sum_{i=1}^n \mu_i \xi_i \quad (18)$$

Perubahan Lp pada *dual problem* menghasilkan formula yang sama dengan persamaan (11) sehingga pencarian pembatas bidang terbaik nonlinear dilakukan dengan cara yang hampir sama dengan kasus linear tetapi rentang nilai α_i adalah $0 \geq \alpha_i \geq C$. Instance yang bernilai $\alpha_i = C$ disebut *bounded support vector*. Metode lain untuk mengklasifikasikan masalah data *non linear* adalah dengan cara mentransformasikan data ke dalam dimensi ruang fitur (*feature space*) sehingga dapat dipisahkan secara linear pada ruang fitur.

Transformasi data pada *feature space* dilakukan dengan cara data dipetakan menggunakan fungsi pemetaan (transformasi) $x_k \rightarrow \phi(x_k)$ ke dalam

feature space sehingga terdapat bidang pemisah yang dapat memisahkan data sesuai dengan kelasnya (gambar 3.4). Contoh terdapat set dengan dua atribut dan dua kelas yaitu positif dan negatif. Data dari kelas positif $\{(2,2),(2,-2),(-2,2),(-2,-2)\}$ dan data dari kelas negatif $\{(1,1),(1,-1),(-1,1),(-1,-1)\}$.



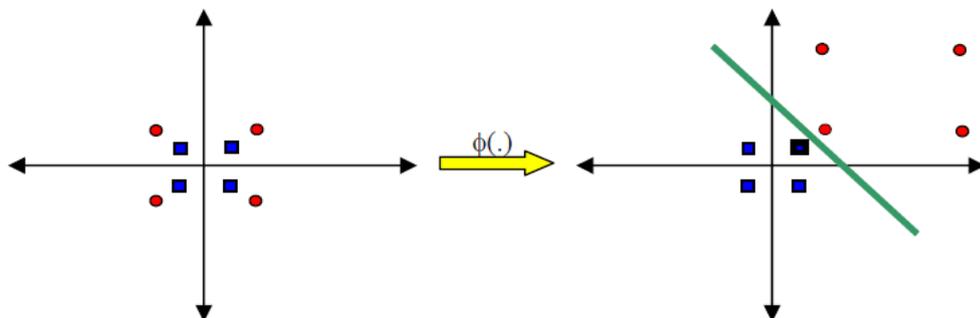
Sumber : (Sembiring, 2007)

Gambar 3. 4 : Transformasi dari vektor input ke feature space

Kemudian data digambarkan pada ruang dimensi dua (gambar 3.5) menghasilkan gambar data tidak dapat dipisahkan secara linear, sehingga digunakan transformasi berikut :

$$\phi(x_1, x_2) = \begin{cases} \sqrt{x_1^2 + x_2^2} > 2 \rightarrow (4 - x_2 + |x_1 - x_2|, 4 - x_1 + |x_1 - x_2|) \\ \sqrt{x_1^2 + x_2^2} \leq 2 \rightarrow (x_1, x_2) \end{cases} \quad (19)$$

Diperoleh data setelah ditransformasi untuk kemudian dilakukan pencarian bidang pemisah terbaik adalah $\{(2,2),(6,2),(6,6),(2,6)\}$ untuk kelas negatif dan untuk kelas positif adalah $\{(1,1),(1,-1),(-1,1),(-1,-1)\}$.



Gambar 3. 5 : Transformasi Data Tidak Dapat Dipisahkan Secara Linier

Dengan fungsi transformasi $x_k \rightarrow \phi(x_k)$, maka didapatkan nilai $w = \sum_{i=1}^{ns} \alpha_i y_i \phi(x_i)$ dan fungsi hasil latih adalah :

$$f(x_d) = \sum_{i=1}^{ns} \alpha_i y_i \phi(x_i) \phi(x_d) + b \quad (20)$$

Feature space pada kenyataanya memiliki dimensi yang lebih tinggi dari pada vektor input (*input space*). Hal ini berakibat komputasi pada *feature space* sangat besar sehingga ada kemungkinan *feature space* memiliki jumlah *feature* tak terhingga. Untuk mengatasi permasalahan ini pada SVM digunakan “*kernel trick*”. Dari persamaan (18) terdapat dot product $\phi(x_i) \phi(x_d)$. Jika terdapat fungsi *kernel* K sehingga $K(x_i, x_d) = \phi(x_i) \phi(x_d)$, maka fungsi $\phi(x_k)$ tidak perlu diketahui. Sehingga diperoleh fungsi dari pelatihan :

$$f(x_d) = \sum_{i=1}^{ns} \alpha_i y_i K(x_i, x_d) + b \quad (x_i = \text{support vector}) \quad (21)$$

dimana :

- $\alpha_i =$ Koefisien *lagrange*
- $x_i =$ Support vector
- $y_i =$ Kelas data
- $ns =$ Jumlah Support vector
- $x_d =$ Data yang diklasifikasikan

Syarat fungsi untuk bisa menjadi fungsi *kernel* adalah memenuhi teorema *Mercer* yang menyatakan bahwa matriks kernel harus bersifat positif *semi-definite*. Fungsi kernel yang umum digunakan adalah :

a. *Kernel Linear*

$$K(x_i, x) = x_i^T x \quad (22)$$

b. *Polynomial Kernel*

$$K(x_i, x) = (\gamma \cdot x_i^T x + r)^p, \gamma > 0 \quad (23)$$

c. *Radial Basis Function (RBF)*

$$K(x_i, x) = \exp(-\gamma |x_i - x|^2), \gamma > 0 \quad (24)$$

d. *Sigmoid Kernel*

$$K(x_i, x) = \tanh(\gamma x_i^T x + r) \quad (25)$$

3.8 Evaluasi Sistem Klasifikasi

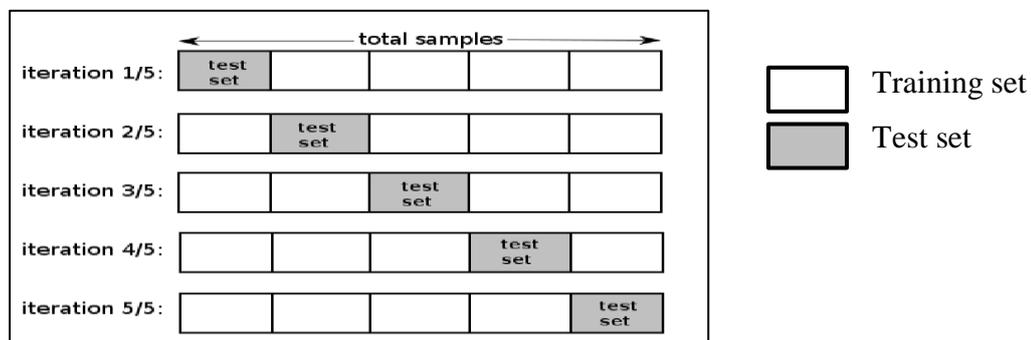
Sebuah sistem klasifikasi harus dinilai performanya agar dapat mengukur tingkat akurasi dari prediksi klasifikasi yang dihasilkan. Ada beberapa metode perhitungan yang digunakan untuk menilai performa sebuah klasifikasi misalnya *K-Fold Cross validation*, *Precision*, *Recall*, dan lain-lain.

3.8.1 K-Fold Cross Validation

Cross validation adalah salah satu teknik untuk memvalidasi keakuratan sebuah model baru yang dibangun berdasarkan dataset baru. Data yang digunakan dalam pembuatan model baru disebut data latih dan data yang digunakan untuk memvalidasi adalah data tes. Model klasifikasi baru dibuat untuk memprediksi suatu dataset baru agar bisa diterapkan dimasa mendatang, sehingga diperlukan suatu validasi agar model yang dihasilkan mempunyai performa yang baik.

K Fold Cross Validation adalah salah satu metode *cross validation* yang digunakan untuk menghitung akurasi prediksi suatu sistem. Data dibagi menjadi k segmen yang memiliki rasio yang sama atau hampir sama. Pelatihan terhadap data dan validasi sebanyak k kali dengan setiap percobaan mengambil satu segmen berbeda sebagai data tes atau validasi dan k-1 segmen lainnya sebagai data latih untuk diambil rata-rata dari hasil tiap iterasi (Refaeilzadeh, Tang, & Liu, 2009).

Sebagai gambaran tentang metode *K-fold cross validation* dapat dilihat pada gambar 3.6 dengan k=5 :



Sumber : (Dyoni, 2017)

Gambar 3. 6 : Ilustrasi 5-Fold Cross Validation

Pada evaluasi klasifikasi terdapat empat kemungkinan yang bisa terjadi dari hasil klasifikasi suatu data. Jika data positif dan diprediksi positif maka akan dihitung sebagai *true positive* dan jika data positif diprediksi negatif maka akan dihitung sebagai *false negative*. Pada data negatif jika diprediksi negatif akan dihitung sebagai *true negative* dan jika diprediksi positif maka akan dihitung sebagai *false positive* (Fawcett, 2006).

Tabel 3. 2 : Confusion Matrix
Aktual

	Class	Positive	Negative
Prediksi	Positive	True Positive (TP)	False Positive (FN)
	Negative	False Negative (FN)	True Negative (TN)

3.8.2 Precision

Precision adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban sistem, precision dapat dihitung dengan persamaan (26).

$$\text{Precision} = \frac{\text{relevant item retrived}}{\text{retrieved item}} = \frac{TP}{TP + FP} \quad (26)$$

3.8.3 Recall

Recall adalah salah satu perhitungan keakuratan prediksi yang digunakan sebagai ukuran tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi, *recall* dapat dihitung melalui persamaan (27).

$$\text{Recall} = \frac{\text{relevant item retrieved}}{\text{retrieved item}} = \frac{TP}{TP + FN} \quad (27)$$

3.8.4 F-Measure

F-measure adalah pengaruh *relative* hasil kombinasi nilai *precision* dengan nilai *recall*. *F-measure* dapat digunakan untuk mengukur kinerja dari sistem klasifikasi yang merupakan rata-rata harmonis dari *precision* dan *recall*. *F-measure* dapat memberikan penilaian yang lebih seimbang, *F-measure* dapat dihitung dengan persamaan (28).

$$F - \text{measure} = \frac{2 \cdot (\text{recall} \cdot \text{precision})}{(\text{recall} + \text{precision})} \quad (28)$$

3.8.5 Accuracy

Accuracy adalah tingkat kedekatan antara nilai prediksi dengan nilai actual. Jika nilai akurasi tinggi maka sebuah sistem akan semakin bagus dalam melakukan prediksi, *accuracy* dapat dihitung dengan persamaan (29).

$$\text{Accuracy} = \frac{\text{Prediksi data benar}}{\text{Total data}} = \frac{TP + TN}{TP + TN + FP + FN} \quad (29)$$

3.9 Word Cloud

Word cloud adalah representasi visual dari data teks yang digunakan untuk menggambarkan kata kunci pada situs web yang berguna untuk mempecepat memahami istilah yang paling menonjol dan untuk menemukan istilah berdasarkan abjad untuk menentukan keunggulan relatifnya (Wikipedia, 2018).

Word cloud (awan kata) adalah kumpulan kata-kata yang paling banyak muncul dalam data teks yang dianalisis. Kata-kata tersebut terkumpul seperti sebuah gumpalan awan yang berisi kata-kata. Intensitas banyaknya kata yang digunakan, ditunjukkan dengan ukuran huruf pada kata. Semakin besar huruf dari kata yang terdapat di awan kata menunjukkan semakin sering kata tersebut muncul. Tampilan awan kata ini lebih menarik serta cepat untuk menemukan kata-kata yang sering muncul, akan tetapi awan kata memiliki kekurangan yaitu tidak dapat menunjukkan frekuensi kata-kata muncul dalam suatu teks yang dianalisis (Adiyana & Hakim, 2015).

3.10 Asosiasi Kata

Korelasi sering digunakan untuk menyatakan hubungan dua atau lebih variabel yang bersifat kuantitatif, sedangkan istilah asosiasi sering diartikan sebagai keeratan hubungan antar dua atau lebih variabel yang bersifat kualitatif (Ulwan, 2016).

Pendekatan asosiasi akan digunakan pada penelitian ini untuk mencari hubungan antar teks *review* aplikasi *e-ticketing* sehingga akan didapatkan informasi yang bermanfaat bagi pihak yang terkait.

$$r_{xy} = \frac{n\sum x_i y_i - (\sum x_i)(\sum y_i)}{\sqrt{\{n\sum x_i^2 - (\sum x_i)^2\}\{n\sum y_i^2 - (\sum y_i)^2\}}} \quad (30)$$

dengan

r_{xy} = nilai korelasi antara variabel x dan variabel y

n = banyaknya pasangan data x dan y

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

$\sum y_i$ = jumlah nilai pada variabel y

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

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

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

3.11 Diagram Sebab Akibat (*Cause and Effect Diagram*)

Diagram sebab-akibat adalah suatu diagram yang menggambarkan garis dan simbol-simbol yang menunjukkan hubungan antara penyebab dan akibat suatu masalah, untuk selanjutnya diambil tindakan perbaikan atas masalah tersebut (Besterfield, 2009).

Diagram Sebab Akibat juga dikenal sebagai diagram Ishikawa dan Fishbone diagram karena bentuknya menyerupai tulang ikan. Dimana, setiap tulang mewakili kemungkinan sumber kesalahan. Diagram ini berguna untuk memperlihatkan faktor-faktor utama yang berpengaruh pada kualitas dan mempunyai akibat pada masalah yang kita pelajari (Haizer & Barry, 2014).

BAB IV

METODOLOGI PENELITIAN

4.1 Populasi dan Sampel

Populasi dalam penelitian ini adalah database website *Google Play Store*, yaitu semua data *review* pengguna aplikasi KAI Access dan Tiket.com, sedangkan sampel yang digunakan dalam penelitian ini adalah *review* pengguna aplikasi KAI Access dan Tiket.com pada tanggal 1 Januari 2018 – 20 Maret 2018. *Review* pada aplikasi Tiket.com hanya yang mengandung topik kereta api.

4.2 Variabel dan Definisi Operasional Variabel

Variabel yang digunakan dalam penelitian ini ditampilkan dalam Tabel 4.1 tentang penjelasan dan definisi operasional penelitian :

Tabel 4. 1 : Variabel Definisi Operasional

Variabel	Definisi Operasional Variabel
Rating	Tingkat kepuasan pengguna aplikasi KAI Access dan Tiket.com
Date	Tanggal pengguna aplikasi membuat <i>review</i> /komentar
<i>Review</i>	Isi komentar pengguna aplikasi KAI Access dan Tiket.com

4.3 Jenis dan Sumber Data

Jenis data yang digunakan pada penelitian ini adalah data primer. Data tersebut diperoleh dengan teknik *web scrapping software python 3.6.4* dari halaman situs web *Google Play Store*.

4.4 Metode Analisis Data

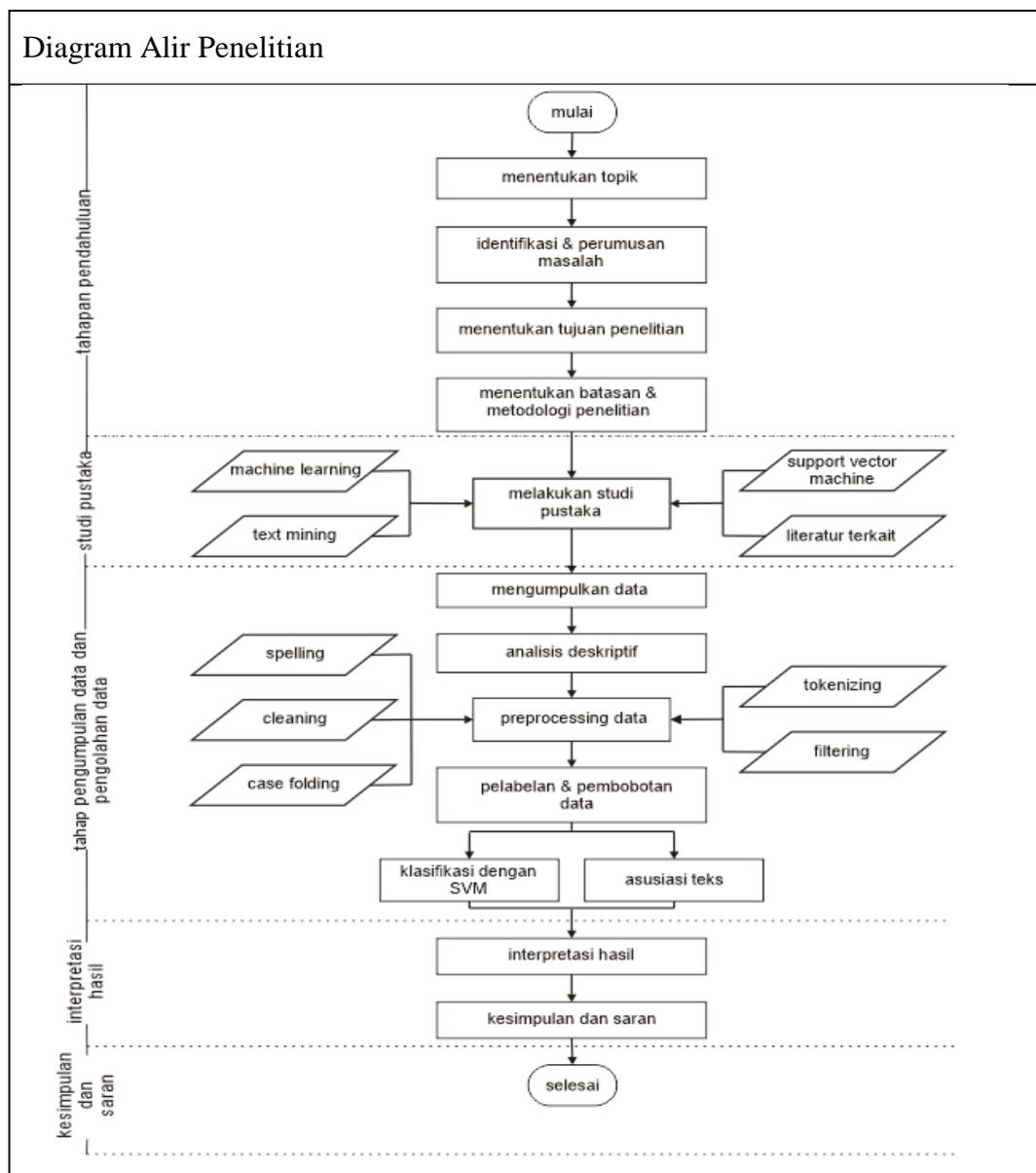
Software yang digunakan dalam penelitian ini adalah *Microsoft Excel 2010*, dan *R Studio*. Ada beberapa metode analisis data yang digunakan dalam penelitian ini antara lain:

- a. Analisis deskriptif, digunakan untuk menggambarkan secara umum persepsi pengguna aplikasi KAI Access dan Tiket.com melalui *review* penggunaannya.

- b. Metode *Machine Learning* yaitu *Support Vector Machine* yang digunakan untuk mengklasifikasi *review* kedalam bentuk positif dan negatif.
- c. *Association* digunakan untuk mengidentifikasi dan membentuk pola kata yang berasosiasi dengan kata lainnya guna mendapatkan informasi yang dianggap penting dan berguna.

4.5 Tahapan Penelitian

Tahapan penelitian yang dilakukan pada penelitian ini digambarkan melalui gambar sebagai berikut:



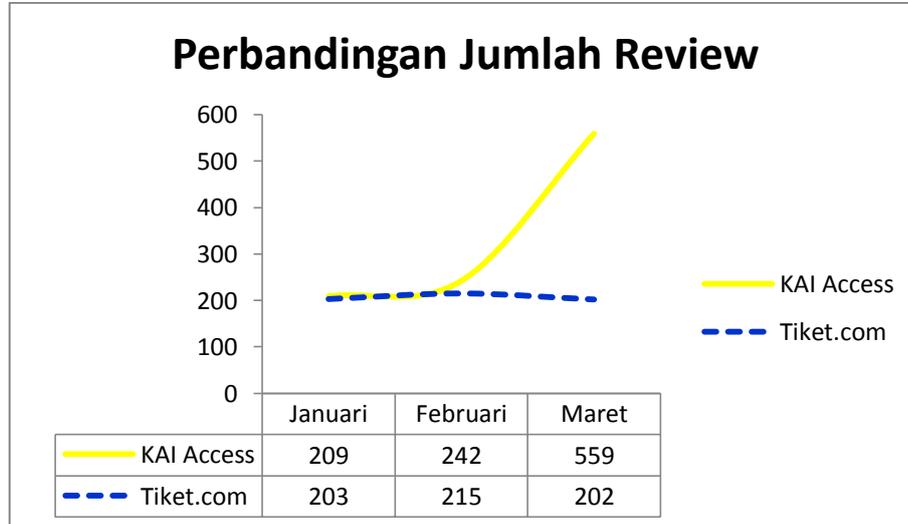
Gambar 4. 1 : Diagram Alir Penelitian

BAB V

PEMBAHASAN

5.1 Analisis Deskriptif

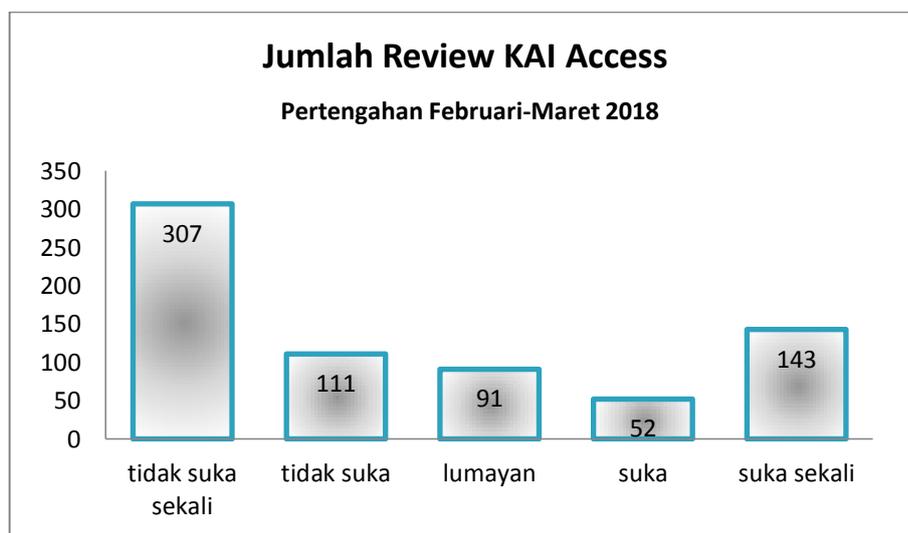
Pertumbuhan jumlah pengguna *smartphone* android mendorong pertumbuhan jumlah unduhan aplikasi di *Google Play Store*. *Play Store* mencatat ada sebanyak 19 miliar unduhan aplikasi pada tahun 2017 (Pertiwi, 2018). Banyaknya aplikasi yang terunduh juga meningkatkan banyaknya pengguna aplikasi berpengaruh juga terhadap banyaknya *review* pada setiap aplikasi, tak terkecuali pada aplikasi *e-ticketing* yang sedang berkembang akhir-akhir ini. Aplikasi *e-ticketing* milik PT Kereta Api Indonesia menjadi sorotan karena aplikasi tersebut merupakan inovasi baru yang dilakukan oleh PT Kereta Api Indonesia. Sebuah inovasi kadang tak selalu sesuai dengan harapan, ada satu dua hal yang harus diperbaiki. Maka dari itu, peneliti akan membandingkan aplikasi KAI Access dan Tiket.com untuk perbaikan pada sistem aplikasi KAI Access.



Gambar 5. 1 : Perbandingan Jumlah Review

Gambaran umum mengenai informasi *review* aplikasi *e-ticketing* Tiket.com dan KAI Access pada tanggal 1 Januari 2018 – 20 Maret 2018 dilakukan dengan menggunakan analisis deskriptif. Jumlah *review* yang masuk dalam sistem *Play Store* dari bulan Januari 2018 sampai dengan pertengahan Maret 2018. Jumlah perbandingan *review* aplikasi KAI Access dan Tiket.com

selama tiga bulan terakhir adalah 62% : 38%. Dari total 1630 *review*, 1010 *review* pada aplikasi KAI Access dan 620 *review* aplikasi Tiket.com. Pada aplikasi KAI Access jumlah *review* cenderung naik pada bulan Maret sedangkan pada aplikasi Tiket.com jumlah *review* cenderung konstan selama tiga bulan tersebut.

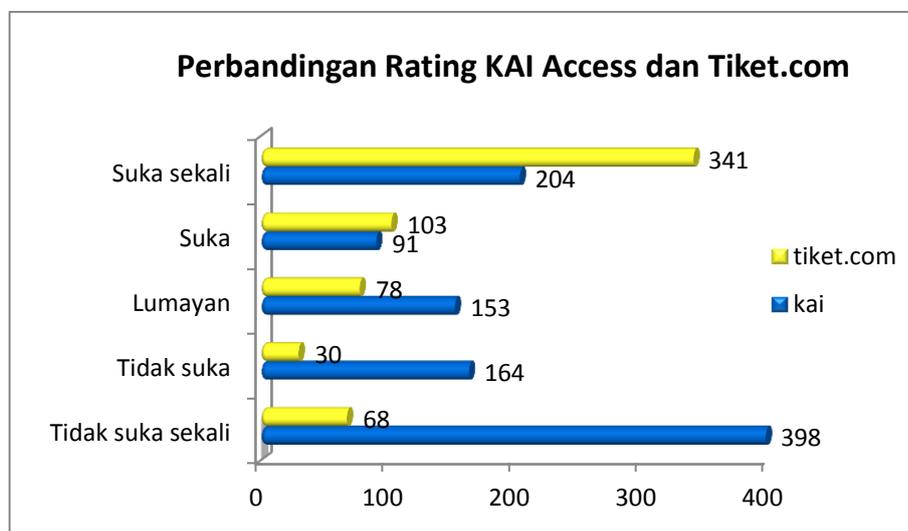


Gambar 5. 2 Jumlah Review KAI Access Feb-Mar

Jika dilihat pada jumlah *review* aplikasi KAI Access pada pertengahan Februari sampai dengan Maret 2018 mengalami kenaikan yang cukup tinggi, *review* tersebut tertinggi sebanyak 307 *review* yang sangat tidak menyukai aplikasi KAI Access, jadi pihak KAI perlu meninjau ulang yang terjadi di bulan Maret 2018 karena banyak pengguna yang merasa kecewa terhadap aplikasi KAI Access sehingga berdampak pada banyaknya *review* negatif yang masuk pada sistem *google play store*.

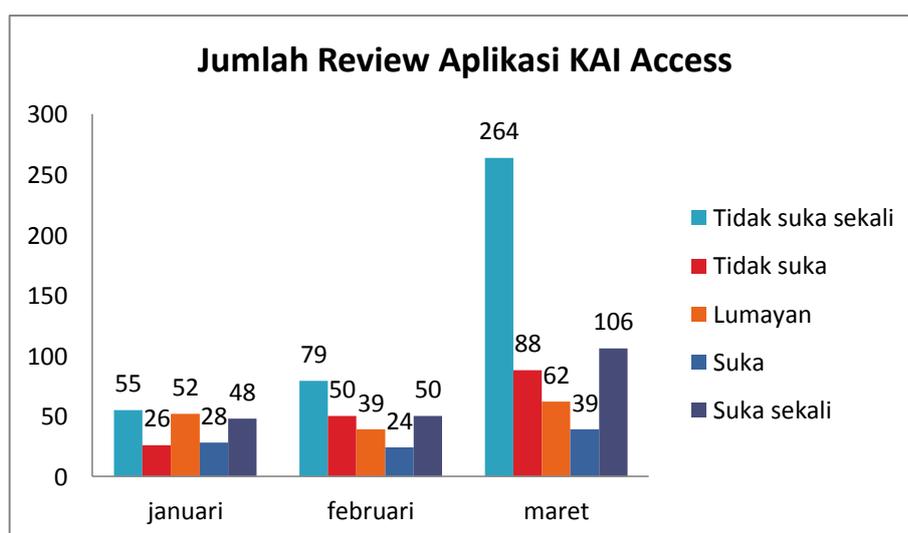
Pada gambar 5.3 menunjukkan perbandingan *rating* pengguna aplikasi KAI Access dan Tiket.com dalam *review* pengguna aplikasi. *Rating* yang diberikan pengguna aplikasi pada situs *Play Store* mempunyai skala 1-5 yang secara berurutan dengan kategori “Tidak suka sekali”, “Tidak suka”, “Lumayan”, “Suka”, dan “Suka sekali”. Dari perbandingan *rating review* pengguna aplikasi Tiket.com sebanyak 341 orang mengaku “Suka sekali” dengankan pengguna aplikasi KAI Access sebanyak 398 orang mengaku “Tidak suka sekali”. Banyaknya jumlah *review* bukan berarti pengguna aplikasi puas akan fitur-fitur

aplikasi tersebut, hal itu dibuktikan walaupun jumlah *review* aplikasi KAI Access lebih banyak namun kebanyakan mengaku “Tidak suka sekali”.



Gambar 5.3 : Perbandingan Rating Aplikasi

Pada gambar 5.4 menunjukkan bahwa banyak sekali pengguna yang merasa “Tidak suka sekali” pada aplikasi KAI Access, terutama pada bulan Maret 2018. Pada Maret 2018 sebanyak 264 orang tidak terpuaskan oleh aplikasi milik PT. Kereta Api Indonesia ini.

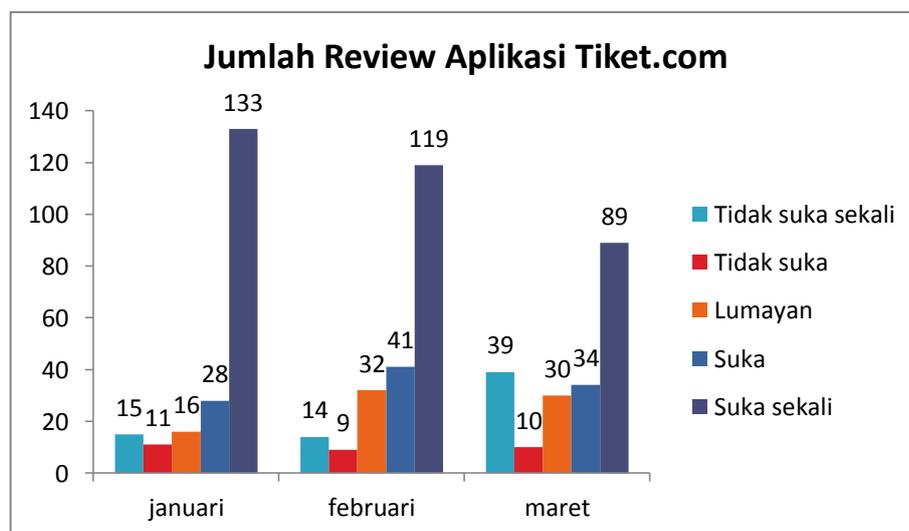


Gambar 5.4 : Jumlah Review Aplikasi KAI Access

Promo yang ditawarkan oleh pihak KAI tak kalah banyak dari pada yang ditawarkan oleh Tiket.com, tetapi permasalahan yang membuat pengguna aplikasi

ini adalah kemungkinan karena setelah pembaharuan pihak KAI tak kunjung melakukan perbaikan. Selain itu pada bulan Maret 2018 banyak pengguna aplikasi kecewa karena tidak memperoleh tiket kereta untuk mudik lebaran.

Gambar 5.5 menunjukkan grafik jumlah *review* aplikasi Tiket.com berdasarkan *rating* yang diberikan pengguna. *Rating* aplikasi Tiket.com pada bulan Januari 2018 terjadi peningkatan rating “Suka sekali” paling tinggi diantara bulan lainnya yaitu sebanyak 133 pengguna. Pada bulan Januari 2018 Tiket.com menawarkan banyak sekali tiket kereta promo awal tahun yang mudah didapatkan oleh pengguna aplikasi yang membutuhkan tiket untuk liburan. Selain itu pada bulan lainnya juga pengguna aplikasi Tiket.com merasa “Suka sekali” tetapi tidak sebanyak pada bulan Januari. Hal ini menunjukkan bahwa aplikasi Tiket.com telah cukup mampu memuaskan pengguna dalam pemesanan tiket kereta api secara *online*.



Gambar 5.5 : Jumlah Review Aplikasi Tiket.com

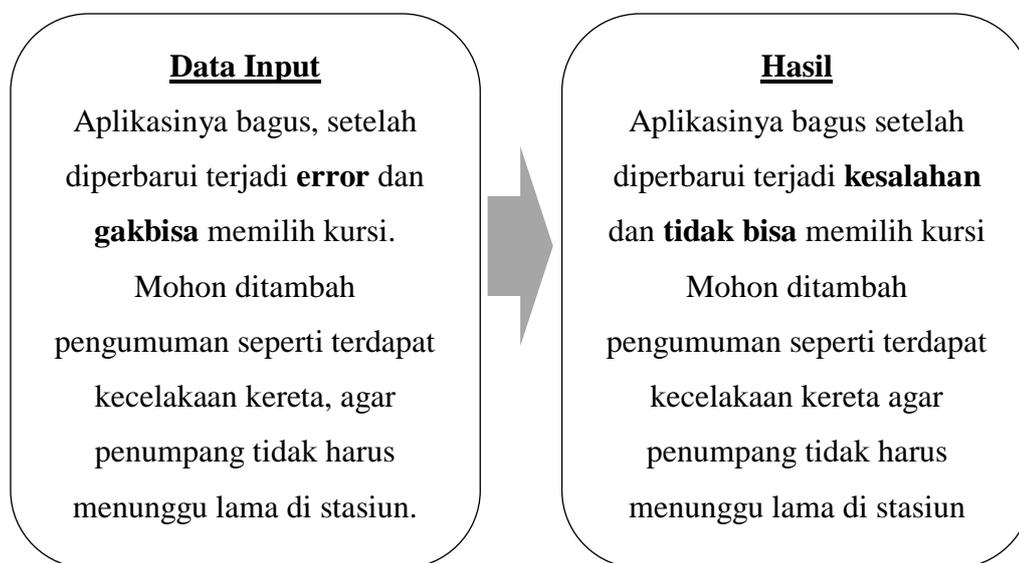
5.2 Teks Preprocessing

Data *review* aplikasi KAI Access dan Tiket.com yang ada pada *website google play store* merupakan data teks yang tidak terstruktur, data teks akan diolah dengan menggunakan metode *text mining* untuk mendapatkan informasi yang dapat berguna bagi berbagai pihak. Proses ini merupakan tahap yang dilakukan untuk mentransformasi data dari yang berbentuk tidak terstruktur

menjadi data yang terstruktur agar mudah untuk dianalisis lebih lanjut. Tahapan yang dilakukan diantaranya adalah *tokenization*, *case folding*, *spelling normalization*, dan *filtering*.

5.2.1 Spelling Normalization

Spelling Normalization merupakan tahap awal pada proses teks yang dilakukan untuk memperbaiki kata-kata yang salah eja atau kata yang disingkat. Misalnya adalah kata “tetapi” memiliki banyak bentuk penulisan kata seperti tapi, tp. Selain itu juga kata “tidak” dapat ditulis dengan tdk, ngga, g, nggak, engga, dan banyak lainnya.



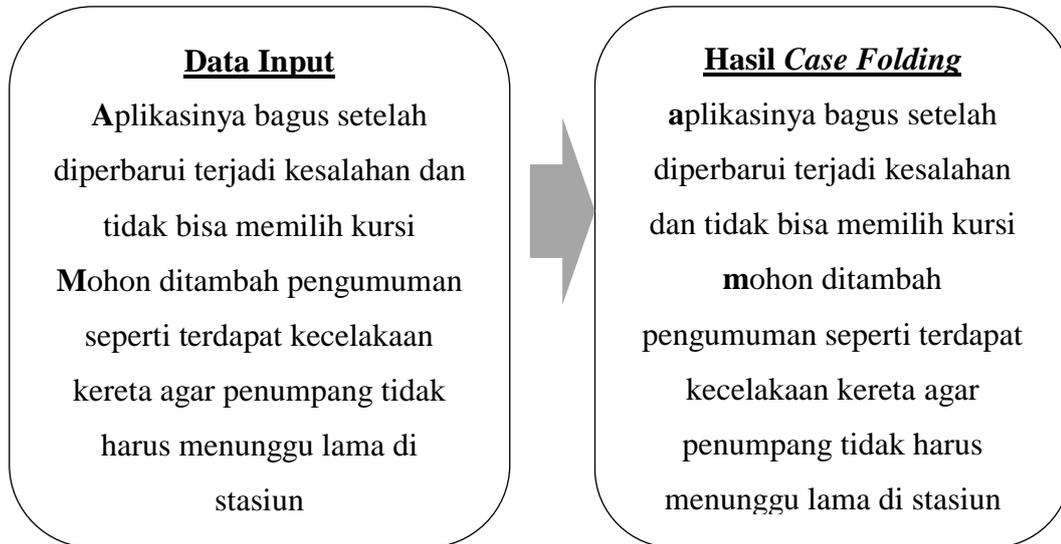
Gambar 5. 6 : Proses Spelling Normalization

Kata yang dicetak tebal “error” akan dirubah menjadi kata “kesalahan” untuk memudahkan perhitungan kelas sentimen pada analisis selanjutnya. Kata “gakbisa” akan dirubah menjadi kata “tidak bisa”, yang merupakan bentuk baku dari arti kalimat tersebut.

5.2.2 Case Folding

Proses merubah huruf menjadi satu bentuk dalam satu dokumen teks, misalnya merubah dari bentuk kapital menjadi huruf kecil atau sebaliknya. Jadi proses ini merupakan penyeragaman bentuk data yang akan di proses pada satu bentuk. Proses *Case Folding* ditunjukkan gambar 5.7. Proses penyeragaman bentuk

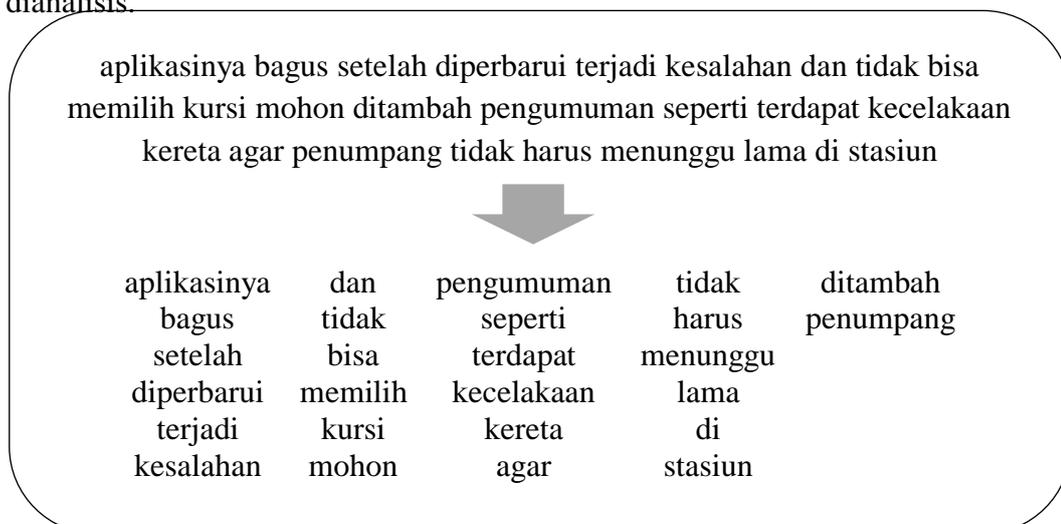
data pada contoh adalah merubah data teks kedalam bentuk huruf kecil, huruf “A” pada kata aplikasinya, huruf “M” pada kata mohon akan dirubah menjadi bentuk huruf kecil untuk menyamakan bentuk teks yang akan dianalisis.



Gambar 5. 7 : Proses Case Folding

5.2.3 Tokenizing

Data teks yang telah seragam atau sama kemudian dilakukan proses *tokenizing*, merupakan proses pemisahan teks menjadi potongan kata yang disebut token. Tujuan dari proses ini adalah mendapatkan potongan kata yang akan menjadi entitas yang memiliki nilai dalam matriks dokumen teks yang akan dianalisis.



Gambar 5. 8 : Proses Tokenizing

Semua data teks akan di pisah menjadi kata per kata yang berperan sebagai token yang akan menjadi entitas bernilai pada proses selanjutnya.

5.2.4 Filtering

Banyaknya *review* aplikasi yang telah terkumpul dari semua pengguna aplikasi KAI Access dan Tiket.com menyebabkan banyaknya frasa dan tanda baca yang tidak seragam serta tidak sesuai sehingga akan menyebabkan performa hasil pengolahan data di tahap selanjutnya tidak maksimal, sehingga kata yang tidak sesuai akan dieliminasi pada proses *filtering*. Kata dan tanda baca yang nantinya tidak bernilai atau tidak berarti akan dieliminasi seperti url, angka, tanda baca, hastag, kata hubung, kata ganti dan lainnya. Pemilihan kata yang bermakna menggunakan algoritma *StopWord* (menghilangkan kata yang kurang penting). Peneliti menggunakan stopword berbahasa Inggris dan berbahasa Indonesia karena bahasa yang digunakan dalam satu *review* aplikasi kadang menggunakan campuran bahasa Inggris dan bahasa Indonesia.



Gambar 5. 9 : Proses Filtering

5.3 Representasi Model

Setelah data melalui tahap *preprocessing*, data perlu dibuat model agar data yang masih berupa kata-kata data diolah dan dihitung. Akan dilakukan perhitungan skor berdasarkan kamus kata untuk kemudian diberi label kelas sentimen. Perhitungan skor sentimen hanya berdasarkan pada jumlah kata yang terdeteksi di kamus *lexicon* dan belum mendeteksi pada tingkatan frasa maupun kalimat.

5.3.1 Perhitungan Skor Sentimen

Berdasarkan *review* pengguna “*aplikasinya bagus setelah diperbarui terjadi kesalahan dan tidak bisa memilih kursi mohon ditambah pengumuman seperti terdapat kecelakaan kereta agar penumpang tidak harus menunggu lama di stasiun*”. Terdapat 3 kata negatif dan 1 kata positif yang terdeteksi pada kamus *lexicon*. Kata negatif yang terdeteksi adalah kesalahan, lama dan menunggu sedangkan satu kata positif yang terdeteksi adalah bagus. Rumus perhitungan skor sentimen yang digunakan pada proses ini adalah jumlah kata positif dikurangkan dengan jumlah kata negatif.

$$\text{Skor} = (\text{jumlah kata positif}) - (\text{jumlah kata negatif}) \quad (31)$$

Tabel 5. 1 : Perhitungan Skor Sentimen

Review	Kata Positif	Kata Negatif
aplikasinya bagus setelah diperbarui terjadi kesalahan dan tidak bisa memilih kursi mohon ditambah pengumuman seperti terdapat kecelakaan kereta agar penumpang tidak harus menunggu lama di stasiun	bagus	kesalahan lama menunggu
Jumlah	1	3
Perhitungan	Skor = 1-3 Skor = -2	

5.3.2 Pelabelan Kelas Sentimen

Analisis sentimen diawali dengan pelabelan data yang dilakukan secara otomatis dengan menghitung skor sentimen menggunakan kamus *lexicon*. Pada tahapan pelabelan akan dilakukan kedalam tiga kelas sentimen yaitu sentimen

positif, sentimen netral dan sentimen negatif. Data akan masuk pada kelas positif jika skor yang didapat > 0 , akan masuk pada kelas negatif jika skor yang di dapat < 0 dan jika skor $= 0$ akan masuk pada kelas netral. Klasifikasi netral didapatkan karena ada dua kemungkinan yaitu :

- a. Tidak terdapat kata sentimen pada data yang terdeteksi pada kamus
- b. Jumlah skor kata positif sama dengan jumlah skor negatif.

Berikut adalah hasil pelabelan kelas sentimen pada *review* aplikasi KAI Acces dan Tiket.com.

Tabel 5. 2 : Hasil Pelabelan Kelas Sentimen

Sentimen	KAI Access	Tiket.com
Positif	216	170
Netral	274	319
Negatif	520	131

Kelas netral akan direduksi karena peneliti hanya akan menggunakan dua pelabelan kelas sentimen yaitu sentimen positif dan sentimen negatif. Pelabelan kelas sentimen netral akan dilakukan secara manual oleh peneliti dengan beberapa pertimbangan yaitu :

- a. Jika pada *review* tidak teridentifikasi kata positif atau negatif maka akan diklasifikasikan pada kelas positif
- b. Jika jumlah kata positif atau negatif sama akan diklasifikasikan sebagai kelas negatif dengan pertimbangan bahwa agar pihak terkait dapat melihat masalah-masalah yang terjadi pada aplikasi setelah aplikasi tersebut di luncurkan dan digunakan oleh masyarakat.

Hasil reduksi pelabelan kelas netral, diperoleh hasil sebagai berikut :

Tabel 5. 3 : Hasil Pelabelan Kelas Sentimen Tanpa Netral

Sentimen	KAI Access	Tiket.com
Positif	280	447
Negatif	730	173
Jumlah	1010	620

Hasil pelabelan kelas sentimen pada kedua aplikasi *e-ticketing* berbeda, pada aplikasi KAI Access jumlah *review* negatif memiliki jumlah yang lebih tinggi dibandingkan dengan jumlah *review* positif sedangkan pada aplikasi Tiket.com sebaliknya *review* positif lebih tinggi dibanding *review* negatif.

5.4 Klasifikasi Sentimen

Setelah kata-kata tersebut diubah menjadi vektor kemudian diberi nilai dan pembobotan untuk setiap kata untuk kemudian diolah menggunakan algoritma prediksi, data tersebut akan dibagi menjadi data latih dan data uji.

5.4.1 Data Latih dan Data Uji

Algoritma klasifikasi menggunakan data latih untuk membentuk model *classifier*, model yang terbentuk adalah hasil representasi pengetahuan yang akan digunakan prediksi kelas data baru yang belum pernah ada sebelumnya. Banyaknya data latih yang digunakan akan menentukan baik tidaknya *machine*/mesin dalam memahami pola data sedangkan data uji digunakan untuk mengukur sejauh mana *classifier* melakukan klasifikasi secara benar. Data latih dan data uji yang digunakan adalah data yang telah memiliki label kelas, dengan perbandingan data uji dan data latih adalah 20% : 80%. Perbandingan tersebut kemudian dapat dilihat pada tabel untuk masing-masing aplikasi *e-ticketing* yaitu KAI Access dan Tiket.com.

Tabel 5. 4 : Pembuatan Data Uji dan Data Latih

Aplikasi	Klasifikasi	Data Latih (80%)	Data Uji (20%)	Jumlah
KAI Access	Positif	224	56	280
	Negatif	584	146	730
	Total	808	202	1010
Tiket.com	Positif	358	89	447
	Negatif	138	35	173
	Total	496	124	620

Pada tabel dapat dilihat bahwa pembuatan data latih data data uji masing-masing aplikasi menggunakan perbandingan yang sama. Jumlah data latih pada aplikasi KAI Access sebanyak 808 *review* sedangkan data uji sebanyak 202

review sehingga jumlah data *review* adalah sebanyak 1010 *review*. Pada aplikasi tiket.com jumlah data latih yang terbentuk adalah sebanyak 496 *review* sedangkan data latih sebanyak 124 *review* sehingga total data yang digunakan adalah sebanyak 620 data *review*.

Pembuatan data latih dan data uji dilakukan dengan pengacakan dengan asumsi bahwa setiap data *review* mempunyai peluang yang sama untuk dapat digunakan sebagai data uji maupun data latih. Pengacakan data dilakukan dengan mempertimbangkan kelas data pada setiap data *review* aplikasi karena proporsi data yang tidak seimbang antara kelas positif dan kelas negatif.

5.4.2 Klasifikasi *Support Vector Machine*

Proses klasifikasi akan dilakukan dengan cara mempelajari data latih yang telah dibentuk pada sub bab sebelumnya. Pada data latih terdapat *review* positif dan *review* negatif dengan perbandingan yang telah ditentukan, kemudian akan dipelajari pola data tersebut dengan algoritma SVM untuk mendeteksi ciri-ciri data pada masing-masing kelas yang telah dibentuk pada data latih. Hasil pelatihan pada algoritma SVM kemudian diuji dengan menggunakan data uji, sehingga didapatkan nilai akurasi dalam prediksi pada data baru. Proses inilah yang disebut sebagai machine learning.

Pada proses klasifikasi menggunakan algoritma SVM metode yang digunakan adalah kernel untuk memperoleh klasifikasi dengan akurasi terbaik. Metode kernel yang digunakan adalah kernel *Linear*, *Polynomial*, *Radial Basis Function* (RBF), dan *Sigmoid*. Percobaan klasifikasi menggunakan data *review* aplikasi KAI Access menghasilkan akurasi tiap metode kernel sebagai berikut :

Tabel 5. 5 : Perbandingan Accuracy Metode Kernel

Kernel	Akurasi
Linear	85.14%
Polynomial	72.27%
Radial	89.60%
Sigmoid	89.10%

Berdasarkan perbandingan hasil beberapa metode kernel, menunjukkan bahwa kernel *Radial Basis Function* (RBS) memiliki tingkat akurasi tertinggi diantara metode kernel yang lainnya. Dengan demikian proses klasifikasi ini akan menggunakan kernel RBS dengan algoritma SVM. Evaluasi model untuk mengetahui hasil akurasi klasifikasi yang terbentuk menggunakan *confusion matrix*. Penelitian ini menggunakan *cross validation* untuk menguji performa *machine learning* dalam membentuk klasifikasi, sehingga terbentuk *5-fold cross validation*. Dari kelima *machine learning* tersebut dipilih nilai akurasi tertinggi dari setiap *review* aplikasi. Hasil masing-masing percobaan *machine learning* menggunakan metode SVM adalah sebagai berikut :

Tabel 5. 6 : Perbandingan Nilai Accuracy Machine Learning

Machine Learning	Akurasi	
	KAI Access	Tiket.com
1	84.60%	84.68%
2	89.60%	78.22%
3	86.13%	75.80%
4	86.13%	80.61%
5	86.63%	83.87%

Terbentuk 5 *machine learning* pada setiap *review* aplikasi, pada klasifikasi *review* KAI Access menghasilkan nilai akurasi tertinggi sebesar 89.60% pada *machine learning* 2 dan pada aplikasi Tiket.com nilai akurasi tertinggi pada *machine learning* 1 sebesar 84.68%.

Tabel 5. 7 : Confusion Matrix

Aplikasi	Prediksi	Aktual	
		Positif	Negatif
KAI Access	Positif	39	17
	Negatif	4	142
	Akurasi	89.60%	
Tiket.com	Positif	87	2
	Negatif	17	18
	Akurasi	84.68%	

Pada *confusion matrix* yang terbentuk menggunakan metode *support vector machine* pada *review* aplikasi KAI Access diperoleh hasil prediksi bahwa pada kelas negatif, dari 159 *review* negative ada sebanyak 142 *review*

terklasifikasikan dengan benar sedangkan sisanya 17 *review* terklasifikasikan masuk dalam kelas positif. Prediksi pada kelas positif *review* aplikasi KAI Access sebanyak 39 *review* terklasifikasikan benar masuk pada kelas positif sedangkan sisanya 4 *review* masuk pada kelas negatif dari total seluruh *review* positif sebanyak 43. Nilai akurasi pada klasifikasi *review* aplikasi KAI Access sebesar 89.60% yang berarti bahwa dari total 202 data *review* sebagai data uji ada sebanyak 181 *review* terklasifikasi dengan benar sedangkan sisanya 21 *review* masuk dalam kelas klasifikasi yang salah.

Pada *review* aplikasi Tiket.com diperoleh hasil prediksi bahwa pada kelas negatif, dari 20 *review* negatif ada sebanyak 18 *review* terklasifikasikan dengan benar sedangkan sisanya 2 *review* terklasifikasikan masuk dalam kelas positif. Prediksi pada kelas positif *review* aplikasi Tiket.com sebanyak 87 *review* terklasifikasikan benar masuk pada kelas positif sedangkan sisanya 17 *review* masuk pada kelas negatif dari total seluruh *review* positif sebanyak 104. Nilai akurasi pada klasifikasi *review* aplikasi Tiket.com sebesar 84.68% yang berarti bahwa dari total 124 data *review* sebagai data uji ada sebanyak 105 *review* terklasifikasi dengan benar sedangkan sisanya 19 *review* masuk dalam kelas klasifikasi yang salah.

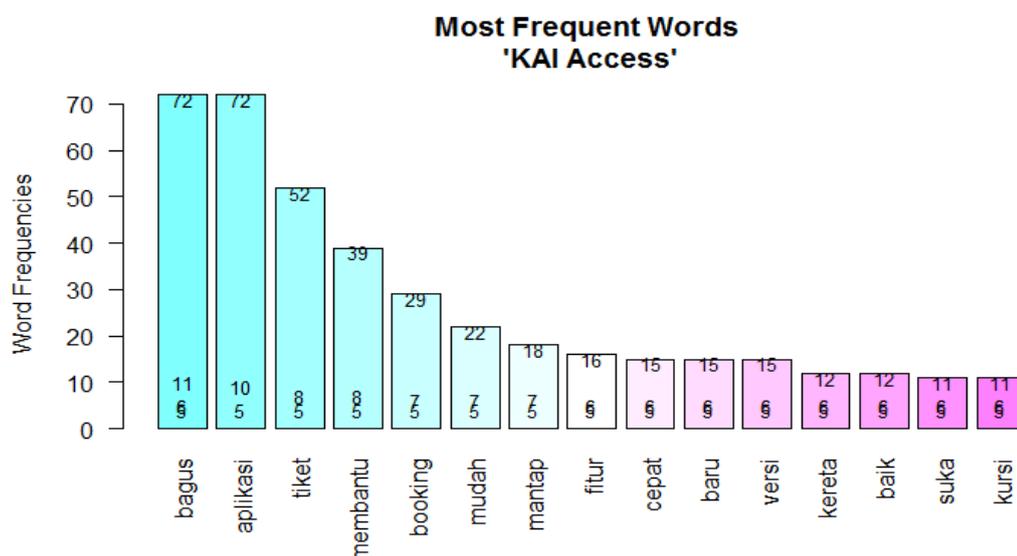
5.5 Visualisasi dan Asosiasi

Visualisasi data *review* dilakukan terhadap aplikasi KAI Access dan Tiket.com dengan tujuan membandingkan sentimen kedua aplikasi tersebut. Aplikasi KAI Access merupakan aplikasi *e-ticketing* milik pemerintah sedangkan Tiket.com merupakan aplikasi milik swasta. Visualisasi *review* kedua aplikasi dilakukan dengan mengekstraksi informasi berupa topik yang sering dibicarakan oleh pengguna kedua aplikasi tersebut. Hasil ekstraksi tersebut berupa informasi yang penting kemudian akan diasosiasikan antar kata yang paling sering muncul secara bersamaan, sehingga mampu memperkuat arti kata dalam suatu informasi. Berikut akan dijelaskan visualisasi berupa diagram frekuensi kata dan *wordcloud* serta asosiasi kata dari setiap *review* aplikasi *e-ticketing*.

5.5.1 KAI Access

Data *review* positif adalah hasil pelabelan yang masuk pada kelas positif menggunakan analisis sentimen berbasis *lexicon*. Ekstraksi *review* yang masuk kelas positif dilakukan berulang kali sampai mendapatkan informasi yang penting untuk menjelaskan tentang aplikasi KAI Acces dan Tiket.com menurut penggunaanya. *Review* positif tersebut diidentifikasi berdasarkan banyaknya frekuensi kata dalam *review*, berikut adalah hasil visualisasi *review* positif dari hasil ekstraksi informasi yang didapatkan dari *review-review* yang ditulis oleh pengguna.

Pada klasifikasi *review* KAI Access, dari jumlah *review* positif berjumlah 280 *review* diperoleh beberapa kata yang paling banyak muncul diantaranya adalah “bagus” dengan frekuensi sebanyak 72 kali, “aplikasi” sebanyak 72, “tiket” sebanyak 52 kali “membantu” sebanyak 39 kali dan seterusnya. Kata dengan frekuensi paling tinggi pada sentimen positif aplikasi KAI Access terdapat pada gambar. Kata – kata tersebut nantinya akan digunakan sebagai dasar penentuan asosiasi kata dengan kata yang lain.



Gambar 5. 10 : Kata Frekuensi Terbanyak dari Kelas Positif KAI Access

Kata dengan frekuensi tertinggi merupakan kata yang paling sering digunakan pada *review* yang ditulis oleh pengguna KAI Access sehingga menunjukkan perasaan pengguna aplikasi yang diwakilkan oleh kata tersebut.

Tabel 5.8 menunjukkan asosiasi kata yang terbentuk pada klasifikasi sentimen positif aplikasi KAI Access, kata-kata tersebut merupakan hal yang paling sering dibicarakan oleh pengguna aplikasi dalam menuliskan *review*nya. Beberapa informasi dari *review* aplikasi KAI Access sebagai berikut :

Kata “aplikasi” berasosiasi dengan kata “fitur”, “bermanfaat”, “lancar”, “keamanan”, dan “tampilannya”. Asosiasi beberapa kata ini menunjukkan beberapa informasi seperti pengguna menyukai aplikasi KAI Access karena adanya fitur baru yang ada pada aplikasi selain itu aplikasi ini bermanfaat untuk pengguna dalam hal pembelian tiket, calon penumpang tidak harus pergi ke stasiun untuk membeli tiket kereta. Aspek kelancaran saat pemesanan tiket dan keamanan sistem aplikasi juga menjadi poin positif yang ada pada aplikasi KAI Access. Tampilan aplikasi KAI Access yang baru juga memanjakan pengguna karena semakin bersahabat dan disukai oleh pengguna.

Kata “kereta” berasosiasi dengan kata “ngopi”, “gratis”, “travelling”, dan “memadai”. Dari kata “ngopi” dan “gratis” dapat diperoleh informasi bahwa PT. Kereta Api Indonesia yang mengadakan program ngopi gratis disambut baik oleh para pengguna aplikasi, selain itu juga menjadi program promosi yang berhasil dalam menarik simpati calon penumpang kereta. Dari kata “Travelling” dan “Memadai” memberikan informasi bahwa transportasi kereta api merupakan sarana transportasi untuk travelling yang memadai, dari aplikasi KAI Access yang bisa untuk memesan tiket kereta sehingga memudahkan pengguna yang akan pergi travelling.

Kata “fitur” berasosiasi dengan kata “menarik”, “joss”, “membantu”, dan “praktis”. Dari keempat kata tersebut dapat diperoleh informasi bahwa fitur yang ada pada aplikasi KAI Access menarik dan joss karena dapat membantu pengguna memesan tiket dengan cepat dan mudah sehingga praktis dalam penggunaannya.

Hasil klasifikasi *review* negatif aplikasi KAI Access sebanyak 730 *review* menghasilkan 15 kata dengan jumlah frekuensi terbanyak ada pada gambar. Kata aplikasi menempati urutan pertama dengan jumlah frekuensi terbanyak sebanyak 231 kali, disusul dengan 96 kali kata “perbaiki” muncul. Hal ini menandakan ada rasa ketidakpuasan pengguna aplikasi KAI Access terhadap performa aplikasi,

banyak sesuatu yang salah pada aplikasi KAIAccess, kemudian di ikuti kata “lama” dan “susah” dengan kata berwarna merah muda. Selain itu pada wordcloud (kanan) merupakan pembentukan tanpa kata”perbaiki”. Kata “error”, “susah” dan “lama” muncul sebagai kata keluhan terhadap aplikasi kemudian kata dengan warna biru tebal muncul yang merupakan kata negatif yang digunakan pengguna aplikasi KAI Access dalam menyampaikan kekecewaannya. Selanjutnya akan dilakukan pencarian asosiasi kata yang sering muncul secara bersama dan diperoleh hasil pada tabel 5.9.

Berdasarkan tabel diperoleh beberapa asosiasi kata pada klasifikasi sentimen aplikasi KAI Access kelas negatif. Proses ekstraksi informasi dilakukan berulang kali sampai menemukan kata-kata yang memiliki hubungan dengan kata lain dan mempertimbangkan relevansi kata dengan topik yang dibicarakan.

Kata “aplikasi” yang mengartikan bagaimana aplikasi KAI Access berjalan pada asosiasi kata yang terbentuk memberikan informasi bahwa aplikasi seperti sampah karena kurangnya persiapan dari developernya dalam pengelolaan aplikasi dan sebagian pengguna menyayangkan kenapa aplikasi ini diluncurkan. Aplikasi KAI Access menjadi hancur dan lama sehingga pengguna aplikasi menjadi malas dan benci. KAI Acces juga dinilai masih kuno dan kurang adanya pembaharuan terhadap sistemnya, sehingga berpengaruh negatif terhadap sentimen pengguna.

Kata “kereta” yang mengartikan keseluruhan fasilitas yang ditawarkan oleh PT KAI melalui aplikasi KAI Access. Asosiasi kata yang terbentuk memberikan informasi bahwa jadwal kereta yang ada pada aplikasi kadang kurang lengkap, saat pengguna mengeluhkan masalah tersebut *customer service* KAI Access kurang tanggap dalam pelayanannya. Selain itu pengguna juga menyarankan agar tiket kereta local dapat dibeli melalui aplikasi sehingga tidak perlu repot dating ke stasiun lagi untuk membeli tiket kereta. Pengguna aplikasi juga menyarankan adanya tambahan kapasitas tiket kereta untuk lebaran karena tiket sudah penuh dan habis.

Kata “kursi” yang mengartikan tempat duduk penumpang kereta api menghasilkan asosiasi kata yang memberikan informasi bahwa pengguna yang

hendak memesan tiket kereta tidak bisa memilih letak kursi sesuai yang diinginkan sehingga pengguna menyarankan agar ada penambahan fitur tersebut dalam aplikasi KAI Access. Selain itu pengguna juga mengeluhkan tampilan kursi yang berantakan dan kadang masih kosong tetapi tidak bisa dipilih.

Tabel 5. 9 : Asosiasi Kata Negatif KAI Access

aplikasi		kereta		kursi	
sampah	0.21	jadwal	0.28	memilih	0.52
developernya	0.17	pelayanannya	0.26	tampilan	0.29
hancur	0.17	antri	0.19	letak	0.29
Kuno	0.15	local	0.19	berantakan	0.28
diluncurkan	0.14	repot	0.14	kosong	0.22
males	0.13	tambahan	0.13	data	
Lama	0.12	kapasitas	0.13	mengisi	0.57
benci	0.11	penuh	0.12	hilang	0.52
busuk	0.10	habis	0.11	menyimpan	0.32
pembayaran		update		sistemnya	0.29
minimarket	0.43	setelah	0.37	tiket	
Kode	0.37	ragu	0.26	mudik	0.46
Valid	0.3	takut	0.2	habis	0.37
metode	0.2	login	0.14	online	0.33
Batas	0.19	error	0.12	perkuota	0.23
ditambah	0.13	nyebelin	0.12	calo	0.18
		kembalikan	0.12	transparan	0.17
		sebelumnya	0.1	harganya	0.15
				jeleknya	0.15

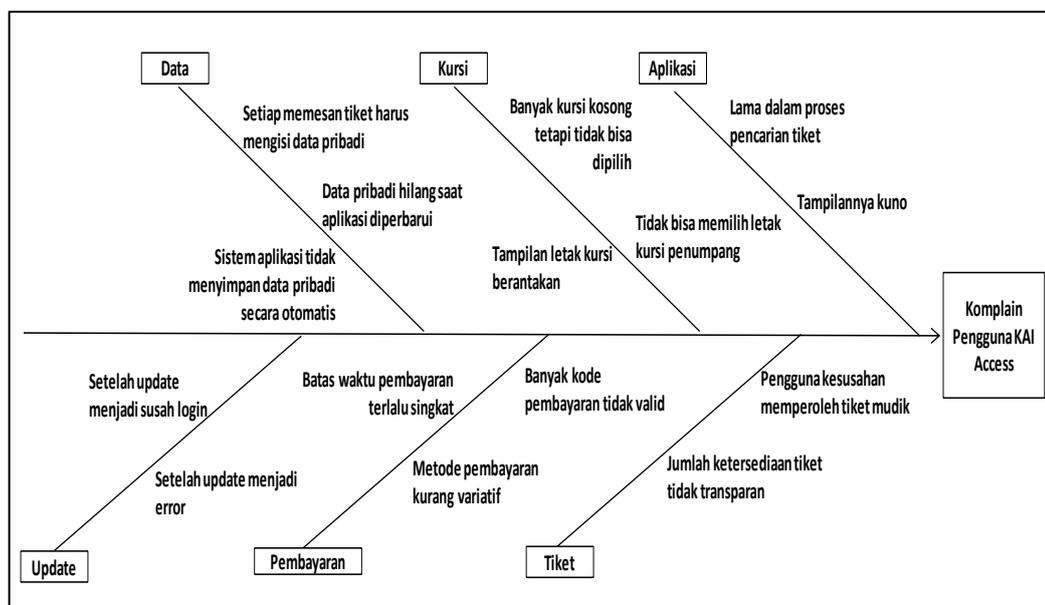
Kata “pembayaran” yang mengartikan metode pembayaran setelah tiket kereta di pesan, asosiasi kata yang terbentuk memberikan informasi bahwa metode pembayaran melalui minimarket kadang masih ada beberapa masalah dan ode pembayaran yang didapatkan oleh pemesan tiket kadang tidak valid. Selain itu pengguna juga mengeluhkan batas waktu pembayaran yang hanya 30 menit setelah pesan, waktu tersebut dinilai masih kurang lama sehingga pengguna meminta tambahan batas waktu pembayaran dan tambahan metode pembayaran agar lebih praktis dalam pembayaran tiket yang sudah dipesan.

Kata “update” yang mengartikan bahwa aplikasi telah diperbaharui dari versi sebelumnya, informasi yang dapat di ambil bahwa pengguna aplikasi merasa ragu dan takut saat akan melakukan pembaharuan aplikasi karena pengguna lain berpendapat bahwa setelah diperbaharui aplikasi menjadi error saat login sehingga banyak yang merasa sebal, untuk itu pengguna meminta agar aplikasi versi sebelumnya dikembalikan karena pengguna lebih puas dengan aplikasi versi sebelumnya.

Kata “data” yang mengartikan data diri calon penumpang yang digunakan untuk registrasi, informasi yang dapat diambil dari asosiasi yang terbentuk adalah pengisian data penumpang kadang masih bermasalah karena sistemnya yang tidak bisa menyimpan data karena data penumpang hilang saat aplikasi diperbaharui.

Kata “tiket” pada asosiasi kata yang terbentuk memberikan informasi bahwa tiket mudik yang disediakan PT KAI habis dalam waktu yang singkat. Pengguna menyarankan agar penjualan tiket kereta dibuka perkuota agar menghindari adanya calo dan penjualan tiket lebih transparan.

Kemudian asosiasi kata negatif dapat dibentuk diagram sebab akibat pada gambar 5.14 untuk perbaikan-perbaikan pada aplikasi.



Gambar 5. 14 Diagram Sebab Akibat KAI Access

Pada gambar 5.14 dapat diketahui faktor-faktor penyebab aplikasi KAI Access memiliki *review* negatif yaitu dari faktor kursi, data, aplikasi, update,

pembayaran, dan tiket. Langkah selanjutnya yaitu menentukan pemecahan masalah, adapun rencana pemecahan masalah aplikasi KAI Access pada dilihat pada tabel 5.10.

Tabel 5. 10 Rencana Pemecahan Masalah Aplikasi KAI Access

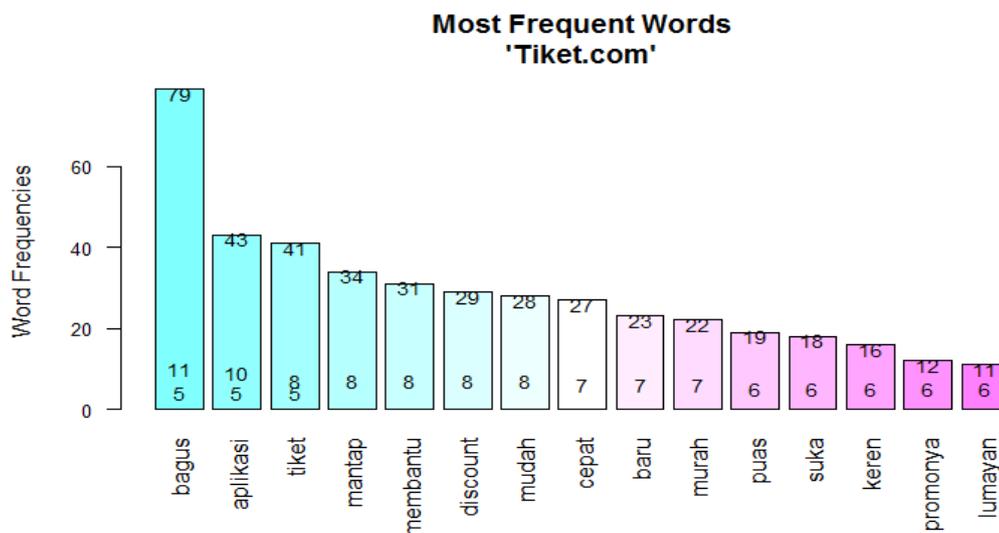
No	Faktor	Permasalahan	Rencana Pemecahan Masalah
1	Aplikasi	Lama dalam proses pencarian tiket	Pihak developer mencari penyebab dan segera melakukan perbaikan
		Tampilan kuno	Pihak developer melakukan pembaruan aplikasi pada sisi tampilan
2	Tiket	Pengguna kesusahan memperoleh tiket mudik	Pihak KAI menyediakan kereta tambahan lebaran
		Jumlah ketersediaan tiket tidak transparan	Melakukan update sisa tiket yang belum terjual
4	Pembayaran	Kode pembayaran tidak valid	Developer melakukan pembaruan sistem aplikasi
		Metode pembayaran kurang variatif	Menambah metode pembayaran seperti m-banking
		Batas waktu pembayaran terlalu singkat	Menambah fitur notifikasi (peringat) sisa waktu pembayaran
5	Data	Setiap memesan tiket harus mengisi data pribadi	Developer melakukan pembaruan sistem aplikasi
		Data pribadi hilang saat aplikasi diperbarui	Developer melakukan backup data pengguna
		Aplikasi tidak menyimpan data pribadi secara otomatis	Developer melakukan pembaruan sistem aplikasi
6	Update	Setelah update susah login	Developer menghilangkan bug-bug pada aplikasi
		Setelah update menjadi error	Developer menghilangkan bug-bug pada aplikasi
7	Kursi	Tidak bisa memilih letak kursi penumpang	Developer melakukan pembaruan sistem aplikasi

No	Faktor	Permasalahan	Rencana Pemecahan Masalah
		Tampilan letak kursi berantakan	Developer melakukan pembaruan sistem aplikasi
		Banyak kursi kosong tetapi tidak bisa dipilih	Developer melakukan pembaruan sistem aplikasi

5.5.2 Tiket.com

Pada hasil klasifikasi positif aplikasi Tiket.com didapatkan 15 kata dengan frekuensi tertinggi seperti pada gambar. Seperti halnya pada *review* positif aplikasi KAI Access, kata “bagus” masih berada pada urutan pertama kata dengan frekuensi tertinggi dengan jumlah 79 kata, di lanjutkan dengan “aplikasi” sebanyak 43 kali, “tiket” sebanyak 41 kali, dan seterusnya. Kata-kata pada gambar merupakan kata terbanyak yang mewakili sentimen positif aplikasi Tiket.com, kata tersebut menjadi topik yang paling banyak dibicarakan oleh pengguna aplikasi Tiket.com. *Review* aplikasi Tiket.com dapat ditampilkan dalam bentuk *wordcloud* yang merupakan kumpulan kata yang paling sering muncul seperti pada gambar 5.16.

Pada *wordcloud* aplikasi Tiket.com gambar 5.16 (kiri) dapat dilihat bahwa kata “bagus” dengan warna merahmuda merupakan kata yang paling banyak banyak digunakan pengguna aplikasi dalam memberikan *review* aplikasi kemudian kata dengan warna merah merupakan kata lanjutan dengan frekuensi yang juga banyak seperti kata “membantu”, “discount”, “mantap” , sedangkan pada gambar 5.16 (kanan) merupakan *wordcloud* tanpa menggunakan kata “bagus” sehingga memunculkan kata positif yang lebih jelas menggambarkan bagaimana Tiket.com dimata para penggunanya. Muncul kata baru berwarna biru tebal yang merupakan kata yang juga banyak digunakan tanpa mempertimbangkan kata “bagus”. Selanjutnya, dilakukan pencarian asosiasi antar kata yang sering muncul secara bersamaan dan diperoleh hasil pada tabel 5.11. Berdasarkan tabel diperoleh beberapa asosiasi kata pada klasifikasi sentimen aplikasi Tiket.com kelas positif. Proses ekstraksi informasi dilakukan berulang kali sampai menemukan kata-kata yang memiliki hubungan dengan kata lain dan mempertimbangkan relevansi kata dengan topik yang dibicarakan.



Gambar 5. 15 : Kata Frekuensi Terbanyak dari Kelas Positif Tiket.com



Gambar 5. 16 : Word Cloud Kelas Positif Tiket.com

Kata “*discount*” berasosiasi dengan kata “banyak” dan ”gede” yang menginformasikan bahwa pihak Tiket.com memberikan banyak *discount* secara besar sehingga membuat pengguna merasa seru dan mantap. Selain potongan harga yang ditawarkan Tiket.com yang murah, pengguna juga merekomendasikan aplikasi ini karena selalu memberikan *discount*. Dari asosiasi kata *discount* yang terbentuk menunjukkan bahwa pemberian *discount* pada strategi pemasaran tiket yang dilakukan aplikasi Tiket.com dapat berpengaruh positif terhadap respon positif para penggunanya.

Tabel 5. 11 : Asosiasi Kata Positif Tiket.com

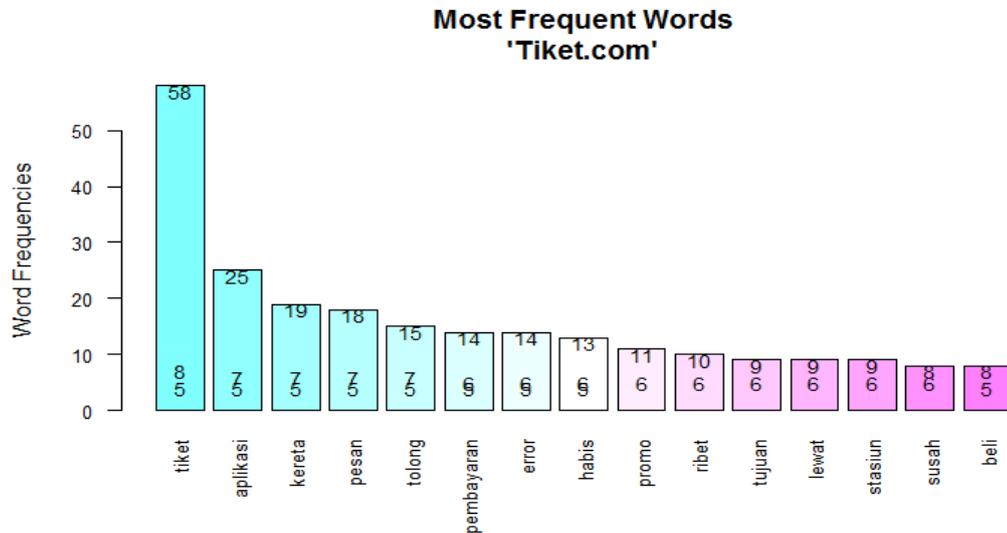
Discount		promo		harga		aplikasi	
banyak	0.46	banyak	0.51	turun	0.55	stabil	0.28
sering	0.39	jelas	0.25	termurah	0.37	pesan	0.20
potongan	0.33	jujur	0.22	lainnya	0.31	tiket	0.19
gede	0.32	stasiun	0.22	jujur	0.27	kereta	0.19
selalu	0.27	e-tiket	0.22	rekomendasi	0.19	cocok	0.19
murah	0.19	rekomendasi	0.15	potongan	0.19	potongan	0.19
harga	0.15	mantap	0.12	discount	0.15	rekomendasi	0.19
seru	0.15	travel	0.12	jelas	0.15	discount	0.15
mantap	0.15	ditambah	0.12	bagus	0.14	murah	0.15
cepat	0.11	harga	0.10	promo	0.10	keren	0.14
rekomendasi	0.10					favorit	0.14
						download	0.12

Kata “promo” yang berarti pemberian program-program menarik dalam penjualan tiket pada aplikasi Tiket.com, memberikan informasi bahwa terdapat banyak promo yang jelas dan jujur pada pembelian tiket kereta api sehingga aplikasi Tiket.com direkomendasikan untuk digunakan karena mantap. Selain itu pengguna meminta agar promo ditambah agar dapat mempermudah jika pengguna akan melakukan travelling.

Kata “harga” yang mengartikan harga tiket yang dijual oleh aplikasi Tiket.com, memberikan informasi bahwa harga tiket di Tiket.com turun dan termurah daripada harga tiket dari aplikasi lainnya. Selain itu harganya yang jujur menjadikan tiket.com direkomendasikan oleh pengguna.

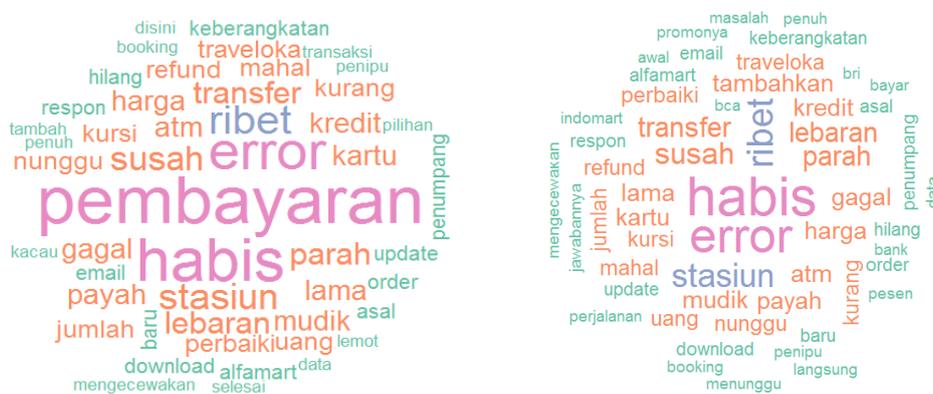
Kata “aplikasi” dengan beberapa asosiasi yang terbentuk memberikan informasi bahwa aplikasi Tiket.com stabil digunakan untuk memesan tiket kereta api. Ditambah dengan adanya *discount*, promo dan potongan harga yang murah menjadikan aplikasi ini difavoritkan oleh penggunanya.

Hasil klasifikasi sentimen kelas negatif aplikasi Tiket.com pada gambar 5.16 menunjukkan 15 kata dengan frekuensi tertinggi terdapat pada gambar. Kata “tiket” menempati urutan pertama dengan jumlah 58 kata digunakan saat membuat sebuah *review* mengenai aplikasi Tiket.com, kemudian ada sebanyak 25 kata “aplikasi, “kereta” digunakan sebanyak 19 kali dan seterusnya.



Gambar 5. 17 : Kata Frekuensi Terbanyak dari Kelas Negatif Tiket.com

Kata tersebut merupakan kata sentimen negatif yang merupakan topik pembicaraan para pengguna aplikasi Tiket.com sebagai dasar menemukan asosiasi dengan kata lainnya. Berikut *wordcloud* yang terbentuk dari sentimen kelas negatif.



Gambar 5. 18 : Word Cloud Kelas Negatif Tiket.com

Pada visualisasi wordcloud dapat di lihat dengan jelas topik yang sering digunakan oleh pengguna aplikasi Tiket.com untuk memberikan *review* negatif. Ukuran kata yang terbentuk menunjukkan banyaknya frekuensi kata tersebut, semakin besar ukuran kata menandakan bahwa kata tersebut sangat sering digunakan pengguna untuk menjelaskan tentang bagaimana aplikasi Tiket.com bekerja. Pada gambar 5.18 (kiri) kata “pembayaran” merupakan kata terbesar

yang terbentuk hal ini menunjukkan bahwa permasalahan yang sering terjadi pada aplikasi Tiket.com adalah soal pembayaran. Kemudian pada gambar 5.18 (kanan) menunjukkan wordcloud tanpa kata “pembayaran”. Selanjutnya akan dilakukan pencarian asosiasi kata yang sering muncul secara bersama dan diperoleh hasil pada tabel 5.12.

Tabel 5. 12 : Asosiasi Kata Negatif Tiket.com

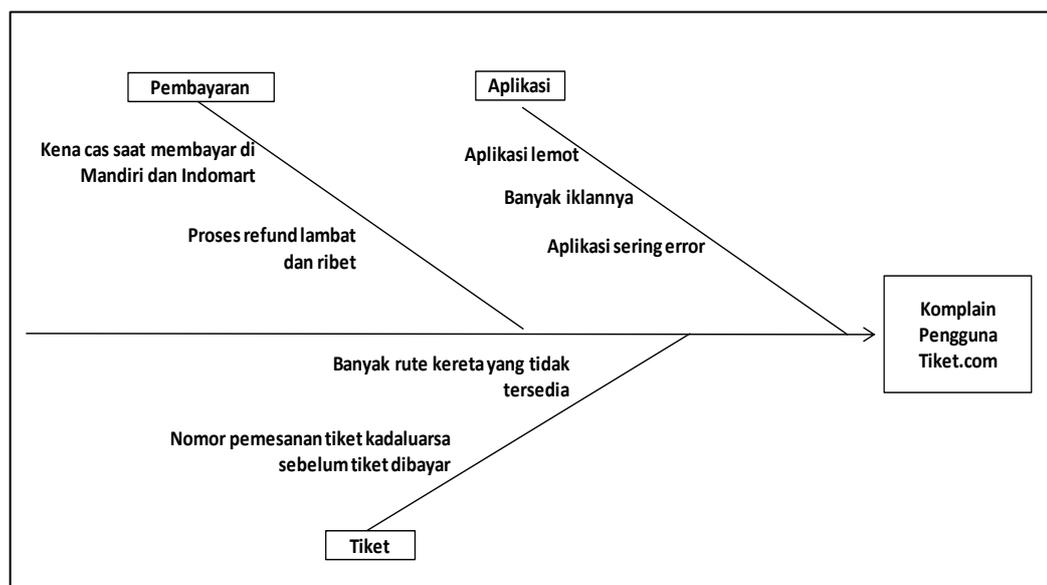
tiket		Aplikasi		pembayaran	
kosong	0.45	parah	0.29	kena	0.50
nomer	0.44	mengecewakan	0.17	cas	0.50
refund	0.37	habis	0.17	mandiri	0.50
error	0.23	penuh	0.17	indomart	0.42
payah	0.20	lemot	0.16	refund	0.28
parah	0.19	error	0.16	lambat	0.24
kadaluarsa	0.11	nyampah	0.16	perbaiki	0.19
nyusahin	0.11	nyesel	0.16	ribet	0.19
kacau	0.11	ilang	0.16		
		taik	0.16		
		iklannya	0.16		
		traveloka	0.11		

Berdasarkan tabel diperoleh beberapa asosiasi kata pada klasifikasi sentimen aplikasi Tiket.com kelas negatif. Proses ekstraksi informasi dilakukan berulang kali sampai menemukan kata-kata yang memiliki hubungan dengan kata lain dan mempertimbangkan relevansi kata dengan topik yang dibicarakan.

Kata “tiket” menghasilkan beberapa asosiasi kata yang menginformasikan tentang banyaknya keluhan pengguna karena tiket kosong. Selain itu pengguna kesusahan saat mendapatkan nomor pesanan tiket sehingga saat akan membayar tiket tersebut *error* dan telah kadaluarsa. Selain itu aplikasi Tiket.com dinilai kacau dan payah karena pengurusan refund uang tiket yang telah dibayarkan bermasalah.

Kata “aplikasi” mengartikan bahwa performa aplikasi Tiket.com membuat kecewa pengguna karena lemot, error dan menjadi banyak iklannya, sehingga banyak yang membandingkan aplikasi Tiket.com dengan aplikasi Traveloka yang lebih baik.

Kata “pembayaran” berarosiasi dengan beberapa kata yang menginformasikan tentang pembayaran yang melalui bank mandiri dan indomart terkena case sehingga dinilai merugikan pengguna. Proses refund pada tiket yang telah dibeli juga lambat dan ribet sehingga pengguna menyarankan agar aplikasi diperbaiki agar lebih baik kedepannya.



Gambar 5. 19 Diagram Sebab Akibat Tiket.com

Kemudian asosiasi kata negatif dapat dibentuk diagram sebab akibat pada gambar 5.18 untuk perbaikan-perbaikan pada aplikasi.

Tabel 5. 13 Rencana pemecahan Masalah Aplikasi Tiket.com

No	Faktor	Permasalahan	Rencana Pemecahan Masalah
1	Aplikasi	Aplikasi lemot	Bagian developer mencari bug aplikasi dan memperbaikinya
		Banyak iklan	Membuat kuis berdasarkan iklan yang menghadiahkan potongan harga tiket.
		Aplikasi sering error	Bagian developer mencari bug aplikasi dan memperbaikinya
2	Tiket	Banyak rute kereta yang tidak tersedia	Melakukan pertemuan dengan pihak KAI terkait penambahan kuota tiket kereta api

No	Faktor	Permasalahan	Rencana Pemecahan Masalah
		Nomor pemesanan tiket kadaluarsa sebelum tiket dibayar	Penambahan fitur notifikasi (peringat) sisa waktu pembayaran tiket.
3	Pembayaran	Kena cas saat membayar di Indomart dan mandiri	Melakukan pertemuan dengan pihak Mandiri dan Indomart terkait pengurangan biaya administrasi
		Proses refund lambat dan ribet	Memperbaiki sistem pelayanan

Pada gambar 5.18 dapat diketahui faktor-faktor penyebab aplikasi Tiket.com memiliki *review* negatif yaitu dari faktor tiket, aplikasi, dan pembayaran. Langkah selanjutnya yaitu menentukan pemecahan masalah, adapun rencana pemecahan masalah aplikasi Tiket.com pada dilihat pada tabel 5.13.

Dari rencana pemecahan masalah yang telah dijabarkan, pihak KAI Access perlu melakukan perbaikan dalam banyak hal khususnya pada performa aplikasi agar server tidak down saat banyak pengguna mengakses pada saat bersamaan, selain itu pihak KAI Access sesekali bisa mencoba mengadakan promo dan discount seperti yang dilakukan oleh pihak Tiket.com.

Pihak Tiket.com perlu melakukan perbaikan dalam sistem pelayanan untuk mendukung sistem aplikasi yang telah baik, selain itu pihak Tiket.com perlu menambah kuota tiket kereta api agar beberapa rute kereta tersedia pada sistem aplikasi Tiket.com.

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan hasil analisis dan pembahasan terhadap data *review* aplikasi KAI Access dan Tiket.com, maka didapatkan beberapa kesimpulan sebagai berikut:

- a. *Rating* aplikasi pada 1 Januari 2018 – 20 Maret 2018, sebanyak 39.4% pengguna sangat tidak menyukai dan 0.20% sangat menyukai aplikasi KAI Access dari total 1010 *review* sedangkan sebanyak 0.10% pengguna sangat tidak menyukai dan 0.55% sangat menyukai aplikasi Tiket.com dari total 620 *review*. Aplikasi Tiket.com lebih unggul daripada KAI Access dalam hal *rating* pengguna.
- b. Hasil klasifikasi sentimen menggunakan metode SVM pada aplikasi KAI Access dengan menggunakan 808 data latih dan 202 data uji. Dari data uji tersebut 159 data masuk kelas negatif dan sisanya 43 data masuk pada kelas positif serta menghasilkan tingkat akurasi sebesar 89.60%, artinya dari 202 *review* terdapat 181 *review* terklasifikasi tepat pada kelas sentimennya. Sedangkan pada aplikasi Tiket.com menggunakan 496 data latih dan 124 data uji. Dari data uji tersebut 20 data masuk kelas negatif dan 104 data masuk kelas positif serta menghasilkan tingkat akurasi sebesar 84.68%, artinya dari 124 *review* terdapat 105 *review* terklasifikasi tepat pada kelas sentimennya.
- c. Secara umum sentimen positif pada aplikasi KAI Access berkaitan dengan adanya ngopi gratis, tampilan dan fitur aplikasi menarik sedangkan pada aplikasi Tiket.com berkaitan dengan banyaknya *discount* dan promo yang menjadikan harga tiket menjadi murah. Sentimen negatif pada aplikasi KAI Access berkaitan dengan errornya aplikasi setelah di update dan susahny mendapatkan tiket kereta untuk lebaran sedangkan pada aplikasi Tiket.com berkaitan dengan susahny refund dan banyaknya iklan pada aplikasi tersebut.

- d. Berdasarkan hasil analisis diagram sebab-akibat (diagram fishbone) terdapat 15 permasalahan hasil identifikasi dari ulasan negatif pengguna KAI Acces dan diklasifikasikan menjadi 6 faktor yang harus diperhatikan yaitu aplikasi, tiket, pembayaran, data, update, dan kursi. Pada aplikasi Tiket.com terdapat 7 permasalahan yang di klasifikasikan menjadi 3 faktor yaitu aplikasi, tiket dan pembayaran.

6.2 Saran

Berdasarkan hasil analisis dan kesimpulan, berikut beberapa saran dari penelitian ini yang mungkin berguna bagi penelitian selanjutnya :

- a. Diharapkan pada penelitian selanjutnya, dataset yang digunakan memiliki perbandingan kelas yang seimbang antara jumlah data kelas positif dan jumlah data kelas negatif.
- b. Pada penelitian ini terbatas hanya review selama tiga bulan, untuk penelitian selanjutnya diharapkan mampu menambah dataset agar hasil penelitian lebih representatif.
- c. Sistem pelabelan kelas sentimen hanya berdasarkan pada susunan kata yang terdapat pada kamus *lexicon*, sehingga kata-kata negasi belum dapat teridentifikasi dengan baik, untuk penelitian selanjutnya sebaiknya dapat menggunakan sistem pelabelan yang memiliki tingkatan lebih tinggi, yakni mampu mendeteksi sentimen pada frasa dan kalimat.
- d. Bagi peneliti selanjutnya, dapat menggunakan pendekatan *machine learning* lain sebagai pembanding performa algoritma *Support Vector Machine*.

DAFTAR PUSTAKA

- Abbas, S. (2000). *Manajemen Transportasi*. Jakarta: Ghalia Indonesia.
- Abtohi, S. (2017). *Implementasi Teknik Web Scrapping dan Klasifikasi Sentiment Menggunakan Metode Support Vector Machine dan Asosiasi*. Skripsi: Program Studi Statistika, Universitas Islam Indonesia.
- Adisasmita, S. A. (2011). *Perencanaan Pembangunan Transportasi*. Yogyakarta: Graha Ilmu.
- Adiyana, I., & Hakim, F. (2015). Implementasi Teks Mining pada Mesin pencarian Twitter Untuk Menganalisis Topik-Topik Terkait KPK dan Jokowi. *Prosiding Seminar Nasional Matematika dan Pendidikan Matematika UMS* (pp. 570-581). Surakarta: UMS.
- Anggraeni, K., & Hidayat, A. A. (2018, Januari 9). *KAI: Pengguna Kereta 389 Juta Orang di Tahun 2017*. Retrieved Maret 24, 2018, from Tempo.co: <https://bisnis.tempo.co/read/1048664/kai-pengguna-kereta-389-juta-orang-di-tahun-2017>
- Besterfield, D. (2009). *Quality Control 8th Edition*. New Jersey: Pearson Prentice Hall.
- Bowersox, C. (1981). *Introduction to Transportation*. New York: MacmillanPublishing Co, Inc.
- Christianini, N., & John S, T. (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge: Cambridge University Press.
- D.Suganthi, & Dr.A.Geetha. (2017). Twitter Sentiment Analisis on GST tweets using R tool. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 793-796.
- Dyoni, P. (2017, Januari 15). *Machine Learning*. Retrieved April 13, 2018, from Medium: <https://medium.com/@piyut.dyoni/machine-learning-buat-yang-ngerasa-bodo-e37bc5b26d9d>
- Fadli, A. (2011). *Konsep Data Mining*. Retrieved April 5, 2018, from IlmuKomputer.com: IlmuKomputer.com

- Fanani, F. (2017). *Klasifikasi Review Software pada Google Play Store Menggunakan Pendekatan Analisis Sentimen*. Skripsi: Ilmu Komunikasi, Universitas Gadjah Mada.
- Fawcett, T. (2006). An Introduction to ROC Analysis. *Pattern Recognition Letters*, 861–874.
- Gunadi, I. A. (2016). *Deteksi Kebohongan Berdasarkan Tanda Emosi Negatif Pada Wajah Dengan Menggunakan Pendekatan Pengklasifikasi Naive Bayesian*. Disertasi: Ilmu Komputer, Universitas Gadjah Mada.
- Gunawan, F., Fauzi, M. A., & Adikara, P. P. (2017). Analisis Sentimen Pada Ulasan Aplikasi Mobile Menggunakan Naive Bayes dan Normalisasi Kata Berbasis Levenshtein Distance (Studi Kasus Aplikasi BCA Mobile). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 1082-1088.
- Haizer, J., & Barry, R. (2014). *Manajemen Operasi*. Jakarta: Salemba.
- Han, J., & Kamber, M. (2012). *Data Mining: Concepts and Techniques Third Edition*. Waltham, MA: Morgan Kaufmann.
- Han, J., Kamber, M., & Pei, J. (2012). *Data Mining Concepts and Techniques*. Waltham: Morgan Kaufmann.
- Hidayatullah, A. F. (2014). *Analisis Sentimen dan Klasifikasi Kategori Terhadap Tokoh Publik pada Data Twitter Menggunakan Naive Bayes Classifier*. Tesis: Ilmu Komputer, Universitas Gadjah Mada.
- Informatikalogi. (2016, November 12). *Pembobotan Kata atau Term Weighting TF-IDF*. Retrieved April 12, 2018, from Informatikalogi.com: <https://informatikalogi.com/term-weighting-tf-idf/>
- KAI. (2016). *Membangun Konektifitas Menggalang Potensi Pertumbuhan*. Annual Report 2016: PT.Kereta Api Indonesia.
- KAIAccess. (2017). *Google Play KAI Access*. Retrieved Desember 27, 2017, from Google Play: <https://play.google.com/store/apps/details?id=com.kai.kaiticketing>
- Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publisher.

- Megawati, C. (2015). *Analisis Aspirasi Dan Pengaduan Di Situs Laporan! Dengan Menggunakan Text Mining*. Skripsi: Program Studi Teknik Industri Fakultas Teknik Universitas Indonesia Depok.
- Miro, F. (2005). *Perencanaan Transportasi Untuk Mahasiswa, Perencana dan Praktisi*. Jakarta: Erlangga.
- Mooney, J. R., & Wong. (2006). Learning for Semantic Parsing with Statistical Machine Translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, (pp. 439-446). New York City, NY.
- Mudikgratis.com. (2017, July 10). *KAI Access, Bukan Sekedar Aplikasi Pemesanan Tiket. Ini Dia Alasannya*. Retrieved Desember 27, 2017, from mudikgratis.com: <http://mudikgratis.com/kai-acces-bukan-sekedar-aplikasi-pemesanantiket-ini-dia-alasannya/>
- Mulajati, M. (2017). *Implementasi Teknik Web Scrapping dan Klasifikasi Sentimen Menggunakan Metode Naive Bayes Classifier dan Asosiasi Teks*. Skripsi: Statistika, Universitas Islam Indonesia.
- Ng-Kruelle, G., Swatman, P., & Kruelle, O. (2006). e-Ticketing Strategy and Implementation in an Open Access System. *Jurnal*, University of South Australia.
- Olabenjo, B. (2016). *Applying Naive Bayes Classification to Google Play Apps categorization*. Jurnal: University of Saskatchewan.
- Pardiyono. (2007). *Pasti Bisa! Teaching Genre-Based Writing Metode Mengajar Writing Berbasis Genre Secara Efektif*. Yogyakarta: Andi Yogyakarta.
- Pertiwi, W. K. (2018, Januari 27). *Google Play Store Catat Rekor Unduhan Aplikasi Terbanyak*. Retrieved Maret 24, 2018, from Kompas.com: <https://tekno.kompas.com/read/2018/01/27/10170077/google-play-store-catat-rekor-unduh-an-aplikasi-terbanyak>
- Pratama, A. H. (2016, Januari 30). *Perkembangan Pengguna Internet di Indonesia Tahun 2016 Terbesar di Dunia*. Retrieved Desember 24, 2017, from techinasia: <https://id.techinasia.com/pertumbuhan-pengguna-internet-di-indonesia-tahun-2016>
- Ramadhani, T. G. (2015). *Analisis Sentimen menggunakan Metode Naive Bayes Classifier dengan Model Dokumen Bernoulli dan Support Vector Machine*. Skripsi: Universitas Gadjah Mada.

- Refaeilzadeh, P., Tang, L., & Liu, H. (2009). *Cross Validation Editors: M. Tamer dan Ling Liu Encyclopedia of Database Systems*. New York: Springer.
- Santosa, R. G. (2004). *Statistik*. Yogyakarta: Andi Offset.
- Santoso, B. (2007). *Data Mining Teknik dan Pemanfaatan Data untuk Keperluan Bisnis*. Yogyakarta: Graha Ilmu.
- Saraswati, N. S. (2011). *Text Mining dengan Metode Naive Bayes Classifier dan Support Vector machines untuk Sentiment Analysis*. Tesis: Program Studi teknik Elektro, Program Pasca Sarjana, Universitas Udayana.
- Sembiring, K. (2007). *Penerapan Teknik Support Vector Machine untuk Pendeteksian Intrusi pada Jaringan*. Skripsi: Program Studi Teknik Informatika Institut Teknologi Bandung.
- Sudrajat, R., Rosadi, R., & Muhammad, H. (2016). Implementation of Data Mining in Analyzing Social Media. *International Journal of Computer Science Issues*, 1694-0814.
- Sugiyono. (2004). *Metode Penelitian*. Bandung: Alfabeta.
- Sujatmiko, W. (2017). *Identifikasi Modus Kejahatan Korupsi Menggunakan Metode naive Bayes Classifier dan Algoritma Apriori*. Tesis: Ilmu Komputer, Universitas Gadjah Mada.
- Suprpto. (2017, Oktober 24). *Sekarang Naik Kereta Api Bisa "Check In" Melalui E-Boarding Pass*. Retrieved Desember 27, 2017, from Kompas.com: <http://travel.kompas.com/read/2017/10/24/101700327/sekarang-naik-kereta-api-bisa-check-in-melalui-e-boarding-pass>
- Tiket.com. (2011). *Tiket.com*. Retrieved April 4, 2018, from Tiket.com: <https://www.tiket.com/faq>
- Ulina D, R. M. (2013). *Pembangun Aplikasi Pencarian Dokumen Menggunakan Text Mining Berbasis Web*. Skripsi: Program Studi Teknik Informatika, Universitas Atma Jaya Yogyakarta.
- Ulwan, M. N. (2016). *Pattern Recognition pada Unstructured Data Teks Menggunakan Support Vector Machine dan Association*. Skripsi: Program Studi Statistika Universitas Islam Indonesia.
- Walpole, R., & Myers, R. H. (1995). *Ilmu Peluang dan Statistika untuk Insinyur dan Ilmuan Terjemahan RK Sembiring*. Bandung: ITB Bandung.

- Wikipedia. (2018, Februari 15). *Google Play*. Retrieved Maret 27, 2017, from Wikipedia Ensiklopedia Bebas: https://id.wikipedia.org/wiki/Google_Play#cite_note-a-4
- Wikipedia. (2018, Januari 26). *Tag Cloud*. Retrieved April 13, 2018, from Wikipedia: https://en.wikipedia.org/wiki/Tag_cloud#Text_clouds
- Wikipedia. (2018, April 11). *Wikipedia* . Retrieved April 13, 2018, from Kereta Api Indonesia: https://id.wikipedia.org/wiki/Kereta_Api_Indonesia
- wikipedia. (2018, Januari 5). *Wikipedia Ensiklopedia Bebas*. Retrieved April 11, 2018, from Pembelajaran Mesin: https://id.wikipedia.org/wiki/Pembelajaran_mesin
- Zaki, M. J., & JR, W. M. (2014). *Data Mining and Analysis Fundamental Concepts and Algorithms*. New York: Cambridge University Press.

LAMPIRAN

Lampiran 1 Script Python Web Scrapping

```
#load webdriver function from selenium
from selenium import webdriver
from time import sleep
from bs4 import BeautifulSoup, Comment
import pandas as pd

#Setting up Chrome webdriver Options
chrome_options = webdriver.ChromeOptions()

#setting up local path of chrome binary file
chrome_path=
r"C:\Users\USER\Documents\chromedriver_win32\chromedriver.exe"

#creating Chrome webdriver instance with the set chrome_options
driver = webdriver.Chrome(chrome_path)
link =
"https://play.google.com/store/apps/details?id=com.kai.kaiticket
ing&hl=in"
driver.get(link)

#driver.execute_script("window.scrollTo(0,
document.body.scrollHeight)")
Ptitle = driver.find_element_by_class_name('id-app-
title').text.replace(' ', '')
print(Ptitle)

#driver.find_element_by_xpath('//*[@id="body-
content"]/div/div/div[1]/div[2]/div[2]/div[1]/div[4]/button[2]/d
iv[2]').click()
sleep(1)
driver.find_element_by_xpath('//*[@id="body-
content"]/div/div/div[1]/div[2]/div[2]/div[1]/div[4]/button[2]/d
iv[2]/div/div').click()

#select_newest.select_by_visible_text('Newest')
driver.find_element_by_xpath('//*[@id="body-
content"]/div/div/div[1]/div[2]/div[2]/div[1]/div[4]/button[2]/d
iv[2]/div/div').click()
sleep(2)

#driver.find_element_by_css_selector('.review-filter.id-review-
sort-filter.dropdown-menu-container').click()
driver.find_element_by_css_selector('.displayed-child').click()

#driver.find_element_by_xpath("//button[@data-dropdown-
value='1']").click()
driver.execute_script("document.querySelectorAll('button.dropow
```

```

n-child')[0].click()")
reviews_df = []
for i in range(1,100):
    try:
        for elem in driver.find_elements_by_class_name('single-
review'):
            print(str(i))
            content = elem.get_attribute('outerHTML')
            soup = BeautifulSoup(content, "html.parser")

#print(soup.prettify())
            date = soup.find('span',class_='review-date').get_text()
            rating = soup.find('div',class_='tiny-star')['aria-
label'][6:7]
            title = soup.find('span',class_='review-title').get_text()
            txt = soup.find('div',class_='review-
body').get_text().replace('Full Review','')[len(title)+1:]
            print(soup.get_text())
            temp = pd.DataFrame({'Date':date,'Rating':rating,'Review
Title':title,'Review Text':txt},index=[0])
            print('-'*10)
            reviews_df.append(temp)

#print(elem)
    except:
        print('s')
        driver.find_element_by_xpath('//*[@id="body-
content"]/div/div/div[1]/div[2]/div[2]/div[1]/div[4]/button[2]/d
iv[2]/div/div').click()
reviews_df = pd.concat(reviews_df,ignore_index=True)
reviews_df.to_csv(Ptitle+'_reviews_kaiaccess.csv',
encoding='utf-8')
#driver.close()

```

Lampiran 2 Script R Preprocessing Data dengan Text Mining KAI Access

```
library(tm)
library(SnowballC)
library(wordcloud)
library(RColorBrewer)
library(stringr)

setwd("/Users/USER/Documents/analisis/")
docs<-readLines("KAI_Access.csv")

# Load the data as a corpus
docs <- Corpus(VectorSource(docs))

#Inspect the content of the document #menampilkan data
#Replacing "/", "@" and "|" with space:
toSpace <- content_transformer(function (x , pattern )
gsub(pattern, " ", x))
docs <- tm_map(docs, toSpace, "/")
docs <- tm_map(docs, toSpace, "@")
docs <- tm_map(docs, toSpace, "\\|")

#Cleaning the text
# Convert the text to lower case
docs <- tm_map(docs, content_transformer(tolower))

#Remove punctuation
docs <- tm_map(docs, toSpace, "[[:punct:]]")

#Remove numbers
docs <- tm_map(docs, toSpace, "[[:digit:]]")

# add two extra stop words: "available" and "via"
myStopwords = readLines("stopwordID.csv")

# remove stopwords from corpus
docs <- tm_map(docs, removeWords, myStopwords)

# Remove your own stop word
# specify your stopwords as a character vector
docs <- tm_map(docs, removeWords,
c("saya","ini","ada","dan","nya","sih",
"jadi","mau","yang","lah","itu","dengan",
"dong","apa","karena","mulu","lagi","aja","kok"))

# Eliminate extra white spaces
docs <- tm_map(docs, stripWhitespace)

# Remove URL
removeURL <- function(x) gsub("http[[:alnum:]]*", " ", x)
docs <- tm_map(docs, removeURL)

#Replace words
docs <- tm_map(docs, gsub, pattern="mesan", replacement="pesan")
```

```

docs <- tm_map(docs, gsub, pattern="seat", replacement="kursi")
docs <- tm_map(docs, gsub, pattern="milih", replacement="memilih")
docs <- tm_map(docs, gsub, pattern="good", replacement="bagus")
docs <- tm_map(docs, gsub, pattern="kalo", replacement="kalau")
docs <- tm_map(docs, gsub, pattern="tiba-tiba",
replacement="tibatiba")
docs <- tm_map(docs, gsub, pattern="sulit", replacement="susah")

#Build a term-document matrix
dtm <- TermDocumentMatrix(docs)
m <- as.matrix(dtm)
v <- sort(rowSums(m),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)
head(d, 10)

#Generate the Word cloud
set.seed(1234)
wordcloud(words = d$word, freq = d$freq, min.freq = 1,
          max.words=95, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))

dataframe<-data.frame(text=unlist(sapply(docs, `[`)),
stringsAsFactors=F)

write.csv(dataframe,
"C://Users/USER/Documents/analisis/KAI_Cleaning.csv")
save.image()

```

Lampiran 3 Script R Preprocessing Data dengan Text Mining Tiket.com

```
library(tm)
library(SnowballC)
library(wordcloud)
library(RColorBrewer)
library(stringr)

setwd("/Users/USER/Documents/analisis/")
docs<-readLines("Tiketdotcom.csv")

# Load the data as a corpus
docs <- Corpus(VectorSource(docs))

#Inspect the content of the document #menampilkan data
#Replacing "/", "@" and "|" with space:
toSpace <- content_transformer(function (x , pattern )
gsub(pattern, " ", x))
docs <- tm_map(docs, toSpace, "/")
docs <- tm_map(docs, toSpace, "@")
docs <- tm_map(docs, toSpace, "\\|")

#Cleaning the text
# Convert the text to lower case
docs <- tm_map(docs, content_transformer(tolower))

#Remove punctuation
docs <- tm_map(docs, toSpace, "[[:punct:]]")

#Remove numbers
docs <- tm_map(docs, toSpace, "[[:digit:]]")

# add two extra stop words: "available" and "via"
myStopwords = readLines("stopwordID.csv")

# remove stopwords from corpus
docs <- tm_map(docs, removeWords, myStopwords)

# Remove your own stop word
# specify your stopwords as a character vector
docs <- tm_map(docs,
removeWords,c("saya","ini","ada","dan","nya","sih","jadi","mau","y
ang","lah","itu","dengan","dong","apa","karena","mulu","lagi","aja
","kok","sangat","klo","atau","tidak","bisa","lebih","sudah","untu
k","kalau","dulu","banget"))

# Eliminate extra white spaces
docs <- tm_map(docs, stripWhitespace)

# Remove URL
removeURL <- function(x) gsub("http[[:alnum:]]*", " ", x)
docs <- tm_map(docs, removeURL)

#Replace words
docs <- tm_map(docs, gsub, pattern="mesan", replacement="pesan")
docs <- tm_map(docs, gsub, pattern="seat", replacement="kursi")
```

```
docs <- tm_map(docs, gsub, pattern="milih", replacement="memilih")
docs <- tm_map(docs, gsub, pattern="good", replacement="bagus")
docs <- tm_map(docs, gsub, pattern="kalo", replacement="kalau")
docs <- tm_map(docs, gsub, pattern="tiba-tiba",
replacement="tibatiba")
docs <- tm_map(docs, gsub, pattern="sulit", replacement="susah")

#Build a term-document matrix
dtm <- TermDocumentMatrix(docs)
m <- as.matrix(dtm)
v <- sort(rowSums(m),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)
head(d, 10)

#Generate the Word cloud
set.seed(1234)
wordcloud(words = d$word, freq = d$freq, min.freq = 1,
          max.words=95, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))

dataframe<-data.frame(text=unlist(sapply(docs, `[`)),
stringsAsFactors=F)

write.csv(dataframe,
"C://Users/USER/Documents/analisis/Tiket_Cleaning.csv")
save.image()
```

Lampiran 4 Script R Pelabelan Kelas Sentimen KAI Access

```
library(tm)
setwd("/Users/USER/Documents/analisis/")
kalimat2<-read.csv("KAI_Cleaning.csv",header=TRUE)

#skoring
positif <-
scan("/Users/USER/Documents/analisis/positive.txt",what="character
",comment.char=";")
negatif <-
scan("/Users/USER/Documents/analisis/negative.txt",what="character
",comment.char=";")
kata.positif = c(positif, "bagus","suka","senang","nice","mantap")
kata.negatif = c(negatif,
"taik","anjing","payah","susah","kecewa")
score.sentiment = function(kalimat2, kata.positif, kata.negatif,
.progress='none')
{
  require(plyr)
  require(stringr)
  scores = laply(kalimat2, function(kalimat, kata.positif,
kata.negatif) {
    kalimat = gsub('[[:punct:]]', '', kalimat)
    kalimat = gsub('[[:cntrl:]]', '', kalimat)
    kalimat = gsub('\\d+', '', kalimat)
    kalimat = tolower(kalimat)
    list.kata = str_split(kalimat, '\\s+')
    kata2 = unlist(list.kata)
    positif.matches = match(kata2, kata.positif)
    negatif.matches = match(kata2, kata.negatif)
    positif.matches = !is.na(positif.matches)
    negatif.matches = !is.na(negatif.matches)
    score = sum(positif.matches) - (sum(negatif.matches))
    return(score)
  }, kata.positif, kata.negatif, .progress=.progress )
  scores.df = data.frame(score=scores, text=kalimat2)
  return(scores.df) }

hasil = score.sentiment(kalimat2$text, kata.positif, kata.negatif)
View(hasil)

#CONVERT SCORE TO SENTIMENT

hasil$klasifikasi<- ifelse(hasil$score<0,"Negatif",
ifelse(hasil$score==0,"Netral","Positif"))
hasil$klasifikasi
View(hasil)

#EXCHANGE ROW SEQUENCE
data <- hasil[c(3,1,2)]
View(data)
write.csv(data, file = "KAI_label.csv")
```

Lampiran 5 Script R Pelabelan Kelas Sentimen Tiket.com

```
library(tm)
setwd("/Users/USER/Documents/analisis/")
kalimat2<-read.csv("Tiket_Cleaning.csv",header=TRUE)

#skoring
positif <-
scan("/Users/USER/Documents/analisis/positive.txt",what="character
",comment.char=";")
negatif <-
scan("/Users/USER/Documents/analisis/negative.txt",what="character
",comment.char=";")
kata.positif = c(positif, "bagus","suka","senang","nice","mantap")
kata.negatif = c(negatif,
"taik","anjing","payah","susah","kecewa")
score.sentiment = function(kalimat2, kata.positif, kata.negatif,
.progress='none')
{
  require(plyr)
  require(stringr)
  scores = laply(kalimat2, function(kalimat, kata.positif,
kata.negatif) {
    kalimat = gsub('[:,punct:]]', '', kalimat)
    kalimat = gsub('[:,cntrl:]]', '', kalimat)
    kalimat = gsub('\\d+', '', kalimat)
    kalimat = tolower(kalimat)
    list.kata = str_split(kalimat, '\\s+')
    kata2 = unlist(list.kata)
    positif.matches = match(kata2, kata.positif)
    negatif.matches = match(kata2, kata.negatif)
    positif.matches = !is.na(positif.matches)
    negatif.matches = !is.na(negatif.matches)
    score = sum(positif.matches) - (sum(negatif.matches))
    return(score)
  }, kata.positif, kata.negatif, .progress=.progress )
  scores.df = data.frame(score=scores, text=kalimat2)
  return(scores.df) }

hasil = score.sentiment(kalimat2$text, kata.positif, kata.negatif)
View(hasil)

#CONVERT SCORE TO SENTIMENT

hasil$klasifikasi<- ifelse(hasil$score<0,"Negatif",
ifelse(hasil$score==0,"Netral","Positif"))
hasil$klasifikasi
View(hasil)

#EXCHANGE ROW SEQUENCE
data <- hasil[c(3,1,2)]
View(data)
write.csv(data, file = "Tiket_label1.csv")
```

Lampiran 6 Script R Klasifikasi menggunakan SVM KAI Access

```
setwd("/Users/USER/Documents/analisis/KAI")
positifL = readLines("PLatih2.txt")
negatifL = readLines("NLatih2.txt")
positifT = readLines("PTest2.txt")
negatifT = readLines("NTest2.txt")

reviewL = c(positifL, negatifL)
reviewT= c(positifT, negatifT)
review_all = c(reviewL,reviewT)
sentiment_training = c(rep("positif", length(positifL) ),
                        rep("negatif", length(negatifL)))
sentiment_test = c(rep("positif", length(positifT) ),
                   rep("negatif", length(negatifT)))
sentiment_all = as.factor(c(sentiment_training, sentiment_test))

library(RTextTools)
library(e1071)

mat = create_matrix(review_all, language = "id", removeStopwords =
FALSE,
                    removeNumbers = TRUE , stemWords = FALSE,
tm::weightTfIdf)
mat = as.matrix(mat)

#SVM
container <- create_container(mat, sentiment_all,
trainSize=1:808,testSize=809:1010, virgin=FALSE)
model <- train_model(container, 'SVM',kernel='radial')
results <- classify_model(container, model)
table(as.character(sentiment_all[809:1010]),
as.character(results[, "SVM_LABEL"]))
recall_accuracy(sentiment_all[809:1010], results[, "SVM_LABEL"])
create_precisionRecallSummary(container, results)
```

Lampiran 7 Script R Klasifikasi menggunakan SVM Tiket.com

```
setwd("/Users/USER/Documents/analisis/Tiketdotcom")
positifL = readLines("PLatih5.txt")
negatifL = readLines("NLatih5.txt")
positifT = readLines("PTest5.txt")
negatifT = readLines("NTest5.txt")

reviewL = c(positifL, negatifL)
reviewT= c(positifT, negatifT)
review_all = c(reviewL,reviewT)
sentiment_training = c(rep("positif", length(positifL) ),
                        rep("negatif", length(negatifL)))
sentiment_test = c(rep("positif", length(positifT) ),
                   rep("negatif", length(negatifT)))
sentiment_all = as.factor(c(sentiment_training, sentiment_test))

library(RTextTools)
library(e1071)

mat = create_matrix(review_all, language = "id", removeStopwords =
FALSE,
                    removeNumbers = TRUE , stemWords = FALSE,
tm::weightTfIdf)
mat = as.matrix(mat)

#SVM
container <- create_container(mat, sentiment_all,
trainSize=1:496,testSize=497:620, virgin=FALSE)
model <- train_model(container, 'SVM',kernel='sigmoid')
results <- classify_model(container, model)
table(as.character(sentiment_all[497:620]),
as.character(results[, "SVM_LABEL"]))
recall_accuracy(sentiment_all[497:620], results[, "SVM_LABEL"])
create_precisionRecallSummary(container, results)
```

Lampiran 8 Script R Visualisasi dan Asosiasi Kata

```
library("tm")
library("SnowballC")
library("wordcloud")
library("RColorBrewer")
library(stringr)
setwd("/Users/USER/Documents/analisis/KAI")
docs<-readLines("KlasNeg.txt")
docs <- Corpus(VectorSource(docs))
# specify your stopwords as a character vector
docs <- tm_map(docs, removeWords,
c("tidak", "bisa", "sudah", "tolong", "malah", "kenapa", "padahal", "buat",
", "udah", "sama", "banget", "juga", "selalu", "dari", "pas", "masih", "bag",
aimana", "selalu", "bikin", "tiba", "saja", "tapi", "saat", "sangat", "moh",
on", "setiap", "sekali", "dulu", "masa", "orang", "harus", "kalau", "lebih",
", "terus", "terus", "setelah", "semua", "sendiri", "min", "jangan", "belu",
m", "untuk", "sering", "bagus", "baru", "atau", "lain", "dapat", "pake", "pa",
kai", "banyak", "seperti", "suka", "close", "pilih", "kali", "kai", "tiket",
", "aplikasinya", "sekarang", "kangkung", "kantor", "molor", "pun", "tak",
", "apalagi", "ampun", "boro", "lo", "sepi", "gak", "gimana", "allah", "ken",
daraan", "bahkan", "mah", "cuma", "banyakan", "janji", "tolonglah", "utam",
akan", "hidup", "cuma", "suruh", "dongg", "gitu"))
docs <- tm_map(docs, gsub, pattern="booking", replacement="pesan")
docs <- tm_map(docs, gsub, pattern="masuk", replacement="login")
docs <- tm_map(docs, gsub, pattern="keluar", replacement="logout")
docs <- tm_map(docs, gsub,
pattern="diperbaiki", replacement="perbaiki")
docs <- tm_map(docs, gsub,
pattern="force", replacement="Forceclose")
docs <- tm_map(docs, gsub,
pattern="sendiri", replacement="Forceclose")
docs <- tm_map(docs, gsub,
pattern="keretanya", replacement="kereta")
docs <- tm_map(docs, gsub,
pattern="ngeceknnya", replacement="ngecek")
docs <- tm_map(docs, gsub, pattern="susahnya", replacement="susah")
docs <- tm_map(docs, gsub,
pattern="aplikasinya", replacement="aplikasi")
docs <- tm_map(docs, gsub, pattern="boking", replacement="booking")
docs <- tm_map(docs, gsub,
pattern="pembayarannya", replacement="pembayaran")
#Build a term-document matrix
dtm <- TermDocumentMatrix(docs)
m <- as.matrix(dtm)
v <- sort(rowSums(m), decreasing=TRUE)
d <- data.frame(word = names(v), freq=v)
head(d, 30)
#Generate the Word cloud
set.seed(1234)
wordcloud(words = d$word, freq = d$freq, min.freq = 1,
max.words=50, random.order=FALSE, rot.per=0.35,
colors=brewer.pal(8, "Dark2"))
#Explore frequent terms and their associations
findFreqTerms(dtm, lowfreq = 4)
#asosiasi kata
```

```
v<-as.list(findAssocs(dtm, terms =c("aplikasi","kereta"),
                             corlimit =
c(0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1)))

head(v)
#barplot
k<-barplot(d[1:20,]$freq, las = 2, names.arg =
d[1:20,]$word,cex.axis=1.2,cex.names=1.2,
          main ="Most frequent words",
          ylab = "Word frequencies",col =topo.colors(20))
termFrequency <- rowSums(as.matrix(dtm))
termFrequency <- subset(termFrequency, termFrequency>=5)
text(k,sort(termFrequency, decreasing = T)-
     1,labels=sort(termFrequency, decreasing = T),pch = 6, cex
=1)
```

Lampiran 9 Stopwords

kata	bagaimanaka	berapalah	bilakah	dibuat
a	h	berapapun	bisa	dibuatnya
ada	bagaimanapu	berarti	bisakah	didapat
adalah	n	berawal	boleh	didatangkan
adanya	bagainamaka	berbagai	bolehkah	digunakan
adapun	h	berdatangan	bolehlah	diibaratkan
agak	bagi	beri	buat	diibaratkanny
agaknya	bagian	berikan	bukan	a
agar	bahkan	berikut	bukankah	diingat
akan	bahwa	berikutnya	bukanlah	diingatkan
akankah	bahwasannya	berjumlah	bukannya	diinginkan
akhir	bahwasanya	berkali-kali	bulan	dijawab
akhiri	baik	berkata	bung	dijelaskan
akhirnya	baiklah	berkehendak	c	dijelaskannya
aku	bakal	berkeinginan	cara	dikarenakan
akulah	bakalan	berkenaan	caranya	dikatakan
amat	balik	berlainan	cukup	dikatakannya
amatlah	banyak	berlalu	cukupkah	dikerjakan
anda	bapak	berlangsung	cukuplah	diketahui
andalah	baru	berlebihan	cuma	diketuahuinya
antar	bawah	bermacam	d	dikira
antara	beberapa	bermacam-	dahulu	dilakukan
antaranya	begini	macam	dalam	dilalui
apa	beginian	bermaksud	dan	dilihat
apaan	beginikah	bermula	dapat	dimaksud
apabila	beginilah	bersama	dari	dimaksudkan
apakah	begitu	bersama-	daripada	dimaksudkan
apalagi	begitukah	sama	datang	nya
apatah	begitulah	bersiap	dekat	dimaksudnya
arti	begitupun	bersiap-siap	demi	diminta
artinya	bekerja	bertanya	demikian	dimintai
asal	belakang	bertanya-	demikianlah	dimisalkan
asalkan	belakangan	tanya	dengan	dimulai
atas	belum	berturut	depan	dimulailah
atau	belumah	berturut-turut	di	dimulainya
ataukah	benar	bertutur	dia	dimungkinka
ataupun	benarkah	berujar	diakhiri	n
awal	benarlah	berupa	diakhirinya	dini
awalnya	berada	besar	dialah	dipastikan
b	berakhir	betul	diantara	diperbuat
bagai	berakhirlah	betulkah	diantaranya	diperbuatnya
bagaikan	berakhirnya	biasa	diberi	dipergunakan
bagaimana	berapa	biasanya	diberikan	diperkirakan
	berapakah	bila	diberikannya	diperlihatkan

diperlukan	guna	janganlah	kelihatan	lihat
diperlukanny	gunakan	jauh	kelihatannya	lima
a	h	jawab	kelima	luar
dipersoalkan	hadap	jawaban	keluar	m
dipertanyaka	hai	jawabnya	kembali	macam
n	hal	jas	kemudian	maka
dipunyai	halo	jelaskan	kemungkinan	makanya
diri	hallo	jelaslah	kemungkinan	makin
dirinya	hampir	jelasnya	nya	maksud
disampaikan	hanya	jika	kena	malah
disebut	hanyalah	jikalau	kenapa	malahan
disebutkan	hari	juga	kepada	mampu
disebutkanny	harus	jumlah	kepadanya	mampukah
a	haruslah	jumlahnya	kerja	mana
disini	harusnya	justru	kesampaian	manakala
disinilah	helo	k	keseluruhan	manalagi
ditambahkan	hello	kadar	keseluruhann	masa
ditandaskan	hendak	kala	ya	masalah
ditanya	hendaklah	kalau	keterlalu	masalahnya
ditanyai	hendaknya	kalaulah	ketika	masih
ditanyakan	hingga	kalaupun	khusus	masihkah
ditegaskan	i	kali	khususnya	masing
ditujukan	ia	kalian	kini	masing-
ditunjuk	ialah	kami	kinilah	masing
ditunjuki	ibarat	kamilah	kira	masuk
ditunjukkan	ibaratkan	kamu	kira-kira	mata
ditunjukkann	ibaratnya	kamulah	kiranya	mau
ya	ibu	kan	kita	maupun
ditunjuknya	ikut	kapan	kitalah	melainkan
dituturkan	ingat	kapankah	kok	melakukan
dituturkannya	ingat-ingat	kapanpun	kurang	melalui
diucapkan	ingin	karena	l	melihat
diucapkannya	inginkah	karenanya	lagi	melihatnya
diungkapkan	inginkan	kasus	lagian	memang
dong	ini	kata	lah	memastikan
dua	inikah	katakan	lain	memberi
dulu	inilah	katakanlah	lainnya	memberikan
e	itu	katanya	laku	membuat
empat	itukah	ke	lalu	memerlukan
enak	itulah	keadaan	lama	memihak
enggak	j	kebetulan	lamanya	meminta
enggaknya	jadi	kecil	langsung	memintakan
entah	jadilah	kedua	lanjut	memisalkan
entahlah	jadinya	keduanya	lanjutnya	memperbuat
f	jangan	keinginan	lebih	memperguna
g	jangan	kelamaan	lewat	kan

memperkirakan	mengingatkan	mungkinah	r	sebetulnya
memperlihatkan	menginginkan	nah	rasa	sebisanya
mempersiapkan	mengira	naik	rasanya	sebuah
mempersoalkan	mengucapkan	namun	rupa	sebut
mempertanyakan	mengucapkan	nanti	rupanya	sebutlah
mempunyai	mengungkapkan	nantinya	s	sebutnya
memulai	mengungkapkan	nyaris	saat	secara
memungkinkan	mengungkapkan	nyata	saatnya	secukupnya
memiliki	menjadi	nyata	saja	sedang
memulai	menjawab	nyatanya	sajalah	sedangkan
memungkinkan	menjelaskan	o	salam	sedemikian
menaiki	menuju	oleh	saling	sedikit
menambahkannya	menunjuk	olehnya	sama	sedikitnya
menandaskan	menunjuki	orang	sama-sama	seenaknya
menanti	menunjukkan	p	sambil	segala
menanti-nanti	menunjukkan	pada	sampai	segalanya
menantikan	menunjuknya	padahal	sampai-	segera
menanya	menurut	padanya	sampai	seharusnya
menanyai	menuturkan	pak	sampaikan	sehingga
menanyakan	menyampaikan	paling	sana	seingat
mendapat	menyangkut	panjang	sangat	sejak
mendapatkan	menyatakan	pantas	sangatlah	sejauh
mendatang	menyebutkan	para	sangkut	sejenak
mendatangi	menyeluruh	pasti	satu	sejumlah
mendatangkannya	menyiapkan	pastilah	saya	sekadar
menegaskan	merasa	penting	sayalah	sekadarnya
mengakhiri	mereka	pentingnya	se	sekali
mengapa	merekalah	per	sebab	sekali-kali
mengatakan	merupakan	percuma	sebabnya	sekalian
mengatakannya	meski	perlu	sebagai	sekaligus
mengenai	meskipun	perlukah	sebagaimana	sekalipun
mengerjakan	meyakini	perlunya	sebagainya	sekarang
mengetahui	meyakinkan	pernah	sebagian	sekaranglah
menggunakan	minta	persoalan	sebaik	sekecil
menghendaki	mirip	pertama	sebaik-	seketika
mengibaratkannya	misal	pertama-tama	baiknya	sekiranya
mengibaratkannya	misalkan	pertanyaan	sebaiknya	sekitar
mengingat	misalnya	pertanyakan	sebaliknya	sekitarnya
	mohon	pihak	sebanyak	sekurang-
	mula	pihaknya	sebegini	kurangnya
	mulai	pukul	sebegitu	sekurangnya
	mulailah	pula	sebelum	sela
	mulanya	pun	sebelumnya	selain
	mungkin	punya	sebenarnya	selaku
		q	seberapa	selalu
			sebesar	selama

selama-	sesuatu	tegas	tunjuk	all
lamanya	sesuatunya	tegasnya	turut	also
selamanya	sesudah	telah	tutur	an
selanjutnya	sesudahnya	tempat	tuturnya	and
seluruh	setelah	tentang	u	another
seluruhnya	setempat	tentu	ucap	any
semacam	setengah	tentulah	ucapnya	are
semakin	seterusnya	tentunya	ujar	as
semampu	setiap	tepat	ujarnya	at
semampunya	setiba	terakhir	umumnya	be
semasa	setibanya	terasa	ungkap	because
semasih	setidak-	terbanyak	ungkapnya	been
semata	tidaknya	terdahulu	untuk	before
semata-mata	setidaknya	terdapat	usah	being
semaunya	setinggi	terdiri	usai	between
sementara	seusai	terhadap	v	both
semisal	sewaktu	terhadapnya	w	but
semisalnya	siap	teringat	waduh	by
sempat	siapa	teringat-ingat	wah	came
semua	siapakah	terjadi	wahai	can
semuanya	siapapun	terjadilah	waktunya	come
semula	sini	terjadinya	walaupun	could
sendiri	sinilah	terkira	walaupun	did
sendirian	soal	terlalu	wong	do
sendirinya	soalnya	terlebih	x	does
seolah	suatu	terlihat	y	each
seolah-olah	sudah	termasuk	ya	else
seorang	sudahkah	ternyata	yaitu	for
sepanjang	sudahlah	tersampaikan	yakin	from
sepantasnya	supaya	tersebut	yakni	get
sepantasnyala	t	tersebutlah	yang	got
h	tadi	tertentu	z	has
seperlunya	tadinya	tertuju	0	had
seperti	tahu	terus	1	he
sepertinya	tak	terutama	2	have
sepihak	tambah	tetap	3	her
sering	tambahnya	tetapi	4	here
seringnya	tampak	tiap	5	him
serta	tampaknya	tiba	6	himself
serupa	tandas	tiba-tiba	7	his
sesaat	tandasnya	tidak	8	how
sesama	tanpa	tidakkah	9	if
sesampai	tanya	tidaklah	\$	in
sesegera	tanyakan	tiga	rp	into
sesekali	tanyanya	toh	about	is
seseorang	tapi	tuju	after	it

its
just
like
make
many
me
might
more
most
much
must
my
never
now

of
on
only
or
other
our
out
over
re
said
same
see
should
since

so
some
still
such
take
than
that
the
their
them
then
there
these
they

this
those
through
to
too
under
up
use
very
want
was
way
we
well

were
what
when
where
which
while
who
will
with
would
you
your