Model *Text Mining* Untuk Identifikasi Keluhan Pelanggan Produk PerusahaanPerangkat Lunak

Rona Neysa Dewi
12917229

Tesis diajukan sebagai syarat untuk meraih gelar Magister Komputer
Konsentrasi Sistem Informasi Enterprise
Program Studi Magister Teknik Informatika
Program Pascasarjana Fakultas Teknologi Industri
Universitas Islam Indonesia
2018

Lembar Pengesahan Pembimbing

Model *Text Mining* Untuk Identifikasi Keluhan Pelanggan Produk Perusahaan Perangkat Lunak

Rona Neysa Dewi
12917229

Yogyakarta, Mei 2018

Pembimbing

Dr.R. Teduh Dirgahayu, ST.,M.Sc

Lembar Pengesahan Penguji

Model *Text Mining* Untuk Identifikasi Keluhan Pelanggan Produk Perusahaan Perangkat Lunak

Rona Neysa Dewi

12917229

Yogyakarta, Mei 2018

Tim Penguji,

Dr. R. Teduh Dirgahayu, ST.,M.Sc

Ketua

Taufiq Hidayat, ST.,MCS

Anggota I

Novi Setiani, ST.,MT

Anggota II

Mengetahui,

Ketua Program Pascasarjana Fakultas Teknologi Industri

Universitas Islam Indonesia

Dr.R. Teduh Dirgahayu, ST.,M.Sc

Abstrak

Model *Text Mining* Untuk Identifikasi Keluhan Pelanggan Produk PerusahaanPerangkat Lunak

Pengguna produk perangkat lunak seringkali menemukan masalah dalam menggunakan produk yang dipakai. Hal itu bisa disebabkan oleh kesalahan produksi atau kesalahan penggunaan. Menghubungi layanan pengaduan atau *customer service* menjadi salah satu jalan dari para pengguna produk untuk mengadukan keluhan yang dialami pengguna.

Jumlah keluhan yang disampaikan kepada customer service sangat beragam dan tercatat dalam *Bugzilla* yang menjadi media penyimpanan keluhan. Dari data keluhan yang sangat beragam itu maka dipisahkan tiap produk. Produk Activity Registration menjadi obyek penelitian ini. Pada perhitungan *precision, recall, dan f-score*diketahui bahwa hasil perhitungan ketiganyamemiliki nilai yang sama dengan perhitungan akurasi. Secara keseluruhan, hasil perolehan *f-score* dengan algoritma Naive Bayesmemberikan hasil yang lebih tinggi dibandingkan dengan algoritma SVM. Model klasifikasi yang dibangun dengan Naïve Bayes memiliki nilai tertinggi ketika menggunakan fitur trigram, sementara model klasifikasi yang dibangun dengan SVM memiliki nilai tertinggi ketika menggunakan fitur unigram.

Kata kunci

bug, text mining, keluhan, customer service, *crisp-dm, svm, naive bayes*

Abstract

Text Mining Model to Identify Customer Product Complaint of Software Company

Users of software products often find problems in using the products they use. It could be due to production errors or misuse. Contacting the complaint service or customer service is one way for users to complain about user complaints.
The number of complaints submitted to customer service is very diverse and recorded in Bugzilla which became the storage media complaints. From the very diverse complaints data then separated each product. Product Activity Registration becomes the object of this research. In the calculation of precision, recall, and f-score note that the results of the three calculations have the same value with the calculation of accuracy. Overall, the f-score yield with the Naive Bayes algorithm provides higher results than the SVM algorithm. The classification model built with Naïve Bayes has the highest value when using the trigram feature, while the classification model built with SVM has the highest value when using the unigram feature.

Keywords

bug, text mining, complain, customer service, crisp-dm, svm, naive bayes

Pernyataan Keaslian Tulisan

Dengan ini saya menyatakan bahwa tesis ini merupakan tulisan asli dari penulis, dan tidak berisi material yang telah diterbitkan sebelumnya atau tulisan dari penulis lain terkecuali referensi atas material tersebut telah disebutkan dalam tesis. Apabila ada kontribusi dari penulis lain dalam tesis ini, maka penulis lain tersebut secara eksplisit telah disebutkan dalam tesis ini.

Dengan ini saya juga menyatakan bahwa segala kontribusi dari pihak lain terhadap tesis ini, termasuk bantuan analisis statistik, desain survei, analisis data, prosedur teknis yang bersifat signifikan, dan segala bentuk aktivitas penelitian yang dipergunakan atau dilaporkan dalam tesis ini telah secara eksplisit disebutkan dalam tesis ini.

Segala bentuk hak cipta yang terdapat dalam material dokumen tesis ini berada dalam kepemilikan pemilik hak cipta masing-masing. Apabila dibutuhkan, penulis juga telah mendapatkan izin dari pemilik hak cipta untuk menggunakan ulang materialnya dalam tesis ini.

Yogyakarta, Februari 2018

Rona Neysa Dewi, S.Kom

Daftar Publikasi

Dewi, R.N. 2017. Model Text Mining Untuk Identifikasi Keluhan Pelanggan Produk Perangkat Lunak. *Seminar Riset dan Inovasi Teknologi Universitas Indraprasta PGRI Jakarta.*

Halaman Kontribusi

Tidak ada kontribusi dari pihak lain.

Halaman Persembahan

Halaman ini saya persembahkan untuk semua pihak yang berkontribusi baik dari segi akademis, finansial, maupun psikis.

Kata Pengantar

Alhamdulillahirabbi'alamin. Puji syukur bagi ALLAH SWT yang telah melimpahkan rahmat dan pertolongan-Nya kepada penulis sehingga penulis dapat menyelesaikan TESIS dengan judul **Model *Text Mining* Untuk Identifikasi Keluhan Pelanggan Produk Perusahaan Perangkat Lunak.** Semoga tesis ini bermanfaat bagi penulis, institusi, dan masyarakat luas.

Selanjutnya penulis mengucapkan terimakasih kepada:

a. Bapak Dr. R. Teduh Dirgahayu selaku Direktur Program Pascasarjana Fakultas Teknologi Industri Universitas Islam Indonesia

b. Bapak Dr. R. Teduh Dirgahayu selaku Dosen Pembimbing I

c. Bapak Taufiq Hidayat, ST., MCS selaku Dosen Pembimbing II

d. Para Dosen Program Studi Magister Teknik Informatika yang telah memberi ilmu pengetahuan kepada penulis

e. Bapak Andy Wahyudi selaku Asisten Direktur PT. VMA yang bersedia memberikan ijin untuk melakukan penelitian di PT. VMA

f. Teman-teman SIE Program Studi Magister Teknik Informatika.

Penulis merasa banyak sekali kekurangan dan kelemahan dalam penelitian ini, oleh karena itu segala kritik dan saran senantiasa penulis harapkan dari para pembaca. Akhir kata, semoga penelitian ini bermanfaat bagi para pembaca dan dimanfaatkan sebaik-baiknya.

Yogyakarta, Mei2018

Penulis,

Rona Neysa Dewi

Daftar Isi

Daftar Tabel

Daftar Gambar

BAB 1
Pendahuluan

1.1 Pendahuluan

Kepuasan konsumen atau kepuasan pelanggan adalah salah satu hal yang diharapkan suatu perusahaan ketika produk yang dihasilkan telah dipasarkan, baik berupa barang maupun jasa. Ditambah lagi dengan persaingan ketat antar perusahaan yang menghasilkan produk serupa membuat perusahaan berlomba-lomba menghasilkan produk yang sesuai dengan keinginan konsumen. Akan tetapi, tak selamanya produk yang sampai di tangan pelanggan benar-benar bebas dari masalah.

Seringkali pelanggan menemukan masalah dalam penggunaan produk, baik dikarenakan kesalahan produksi atau kesalahan cara penggunaan. Menurut penelitian Rose Beard ditahun 2014 menunjukkan bahwa 96% pelanggan yang merasa tidak puas dengan suatu produk tidak pernah menyampaikan keluhannya dan 91% dari pelanggan yang tidak puas itu tidak akan menggunakan produk itu lagi (Beard, 2014). Oleh sebab itu, perusahaan membuka komunikasi sebaik mungkin dengan para pelanggan melalui *customer service*. Melalui *customer service,* pelanggan bisa mengadukan beragam keluhan dari produk, bertanya mengenai cara penggunaan maupun memberi saran untuk perbaikan produk itu sendiri apabila kurang memuaskan.

Cara pengaduan melalui *customer service* pun beragam. Belakangan ini tak hanya melalui telepon saja pelanggan menyampaikan saran atau keluhannya. Pelanggan bisa menyampaikan saran atau keluhannya melalui surat elektronik atau *e-mail* maupun forum-forum di dunia maya yang memang dibuat perusahaan penghasil prdouk untuk menampung beragam keluhan, saran, dan kritik langsung dari konsumen. Setelah melalui langkah ini, perusahaan yang diwakili oleh *customer service* mengumpulkan dan menganalisa keluhan dari pelanggan atau konsumen untuk mengambil tindakan pada langkah selanjutnya, yang tentunya sesuai dengan permintaan konsumen atau pelanggan (Sun et. al, 2011).

1.2 Latar Belakang

PT. VMA atau DWC merupakan salah satu *software house* atau perusahaan perangkat lunak yang menghasilkan produk perangkat lunak untuk administrasi sekolah dengan. Pelanggan pengguna terakhir atau *end user* dari produk PT. VMA adalah sekolah-sekolah yang tak jarang menyampaikan beragam keluhan saat menggunakan perangkat lunak hasil produksi perusahaan ini.

Para pelanggan PT. VMA menyampaikan keluhan melalui beberapa alternatif yaitu telepon, aplikasi *chatting* Skype, atau melalui *e-mail* yang ditangani beberapa *customer service*. Setelah *customer service* berkomunikasi dengan pelanggan, maka keluhan tersebut diseleksi untuk disampaikan kepada tim *developer*atau bagian teknis yang menangani langsung secara teknis pembuatan dan perawatan perangkat lunak. Proses penyampaian keluhan pelanggan dari *customer service* ke bagian developer menggunakan *Bugzilla* sebagai aplikasi penyampaian keluhan yang ditunjukkan langsung kepada tim *developer*. *Bug ticket* dibuat oleh *customer service* untuk menyampaikan deskripsi keluhan. *Bug ticket* sendiri adalah istilah yang digunakan pada aplikasi Bugzilla yang memiliki nomor-nomor tertentu yang bisa berisi tugas, komentar, dan sebagainya (Bugzilla Team, 2015). Selama ini kurang lebih 6700 *bug ticket* yang dibuat oleh *customer service*yang berisi keluhan pelanggan produk-produk PT. VMA dan tersimpan di dalam *data base* Bugzilla.Beragam keluhan tersebut memiliki jumlah kata yang sangat banyak.

Jumlah keluhan yang tercatat yang berjumlah begitu besar tersebut dapat didefinisikan sebagai *Big Data*. *Big Data* merupakan data yang mempunyai jumlah dan variasi besar, serta bergerak cepat, sehingga melampaui kapasitas pengolahan *database* konvensional (Dumbill, 2014). Dalam mengolah *Big Data*, *Data Mining* merupakan metode yang dapat mengotomatisasi proses pengolahan data untuk mengekstraksi pengetahuan dari informasi yang tidak bisa diamati hanya dengan melihat data karena terlalu rumit atau multidimensi. Pada kasus data keluhan pelanggan di PT. VMA yang merupakan data teks, jenis metode *Data Mining* yang dapat digunakan adalah *Text Mining*. Text Mining memegang peran penting dalam analisis *Big Data* yang bersifat tidak terstruktur seperti data teks dan dalam jumlah yang sangat besar (Xiang et al, 2015)

Text mining adalah adalah tipe *natural – languange processing* atau pengolahan bahasa alami yang menguraikan istilah (berupa kata dan frasa) dari dokumen tertentu (Gegick et.al, 2009). Manfaat dari *text mining* itu sendiri adalah untuk menghasilkan inovasi yang membantu orang untuk mengerti akan suatu sistem dengan menggunakan gudang dokumen (Kumar, 2009). Hal ini sangat berbeda.

1.3 Rumusan Masalah

Bagaimana cara organisasi mengetahui jenis masalah yang paling sering disampaikan pelanggan melalui *customer service* berdasarkan kalimat atau kata yang paling sering muncul di keluhan pertama pada *bug ticket* Bugzilla.

1.4 Batasan Masalah

Batasan masalah dalam penelitian ini antara lain:

1. Untuk mengelompokkan apa saja yang menjadi prioritas bug yang harus dikerjakan oleh tim teknis (*programmer dan tester*).
2. Data diambil dari *database* sistem pelacakan *bug* Bugzilla.
3. Penelitan ini hanya menganalisa teks keluhan dari pelanggan yang ada pada *bug ticket* yang ditulis oleh *Customer Service (CSR).*
4. Data dalam penelitian ini memakai istilah dalam bahasa Inggris secara keseluruhan.
5. Obyek penelitian adalah Activity Registration atau AR

1.5 Tujuan dan Manfaat Penelitian

Tujuan dari penelitian ini adalah membuat model data keluhan pelanggan di PT. VMA. Setelah model terbentuk maka diharapkan mampu untuk pengklasifikasian data keluhan secara mudah.

1.6 Manfaat Penelitian

Manfaat dari penelitian yang dilakukan yaitu:

1. Dengan adanya penelitan ini diharapkan memberikan informasi pada tim *developer* untuk meningkatkan kualitas sistem yang dibuat.
2. Membantu proses dokumentasi dan administrasi tiap proyek yang masih berlangsung.

3. Mengetahui tingkat pemahaman pelanggan akan produk yang dihasilkan dan dipasarkan perusahaan
4. Penelitian ini diharapkan dapat memberikan kontribusi untuk pengembangan literatur dalam penelitian yang berhubungan dengan kepuasan konsumen.

1.7 Sistematika Penulisan

Untuk memudahkan penelitian, baik proses penelitian maupun pembuatan laporan penelitian, maka dibuatlah sistematika dan runtutan proses penelitian sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi latar belakang penelitian, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, review penelitian, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini membahas landasan-landasan teori terkait *bug tracking tools* yang digunakan oleh *customer service*, klasifikasi dokumen, dan tahapan-tahapan dalam text mining.

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang detail metodologi penelitian mulai dari studi pustaka, pengumpulan data, langkah-langkah CRISP-DM, langkah-langkah *text mining*, langkah untuk memperoleh hasil dari klasifikasi dan pembobotan, serta pembuatan model dan evauluasi.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi pembahasan hasil pemrosesan data dari *text mining*, hasil pembuatan model dan evaluasi.

BAB V

Bab ini berisi kesimpulan dan saran dari hasil penelitian terkait hasil pembuatan model. Selain itu, bab ini juga berisi saran dan masukan untuk kemajuan penelitian selanjutnya.

BAB II

Tinjauan Pustaka

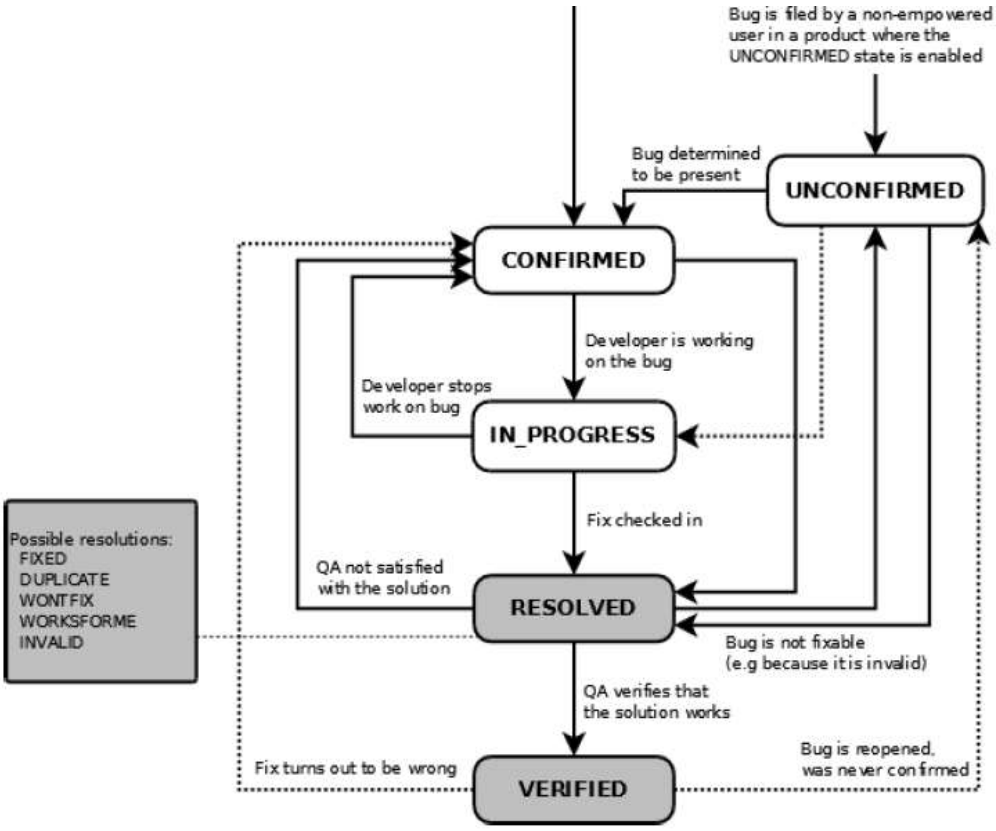
2.1 Bug Tracking Tools

Bug Tracking Tools atau sistem pelacakan bug adalah alat berupa perangkat lunak yang digunakan oleh tim pengembang untuk melacak masalah yang ada pada perangkat lunak atau yang biasanya disebut bug. Sebuah sistem pelacakan bug digunakan untuk menyimpan informasi tentang bug yang dilaporkan.

Di dalam *bug tracking tools* terdapat *database* yang menyimpan data – data yang berkaitan dengan masing – masing *bug* (Bugzilla Documentation, 2015). Data- data itu antara lain waktu *bug* itu dilaporkan, tingkat kerusakan, siapa saja yang membuat bug, tenggang waktu yang diperlukan untuk memperbaiki *bug*, apa saja yang harus dilakukan *programmer* untuk melakukan perbaikan.Sistem pelacakan bug ada yang bersifat *open source* maupun berbayar.

2.2 Bugzilla

Bugzilla adalah salah satu sistem pelacakan bug yang bersifat *open source* yang memungkinkan individu atau kelompok pengembang untuk melacak bug/kesalahan yang ada dalam produk mereka secara efektif. Beragam aktifitas dalam Bugzilla antara lain melacak bug dan kode perubahan, mengelola jaminan kualitas dari suatu *software* yang sedang dibangun dan berkomunikasi antar sesama anggota tim (*Bugzilla Documentation, 2015*). Sistem ini dibangun dengan menggunakan bahasa pemrograman Perl dan sudah digunakan dalam pengembangan sistem yang bersifat open source juga seperti Linux, GNOME, dan Apache (Natarajan, 2010). Siklus status *bug ticket* dalam Bugzilla tergambarkan seperti berikut ini (*Bugzilla Documentation*, 2015).

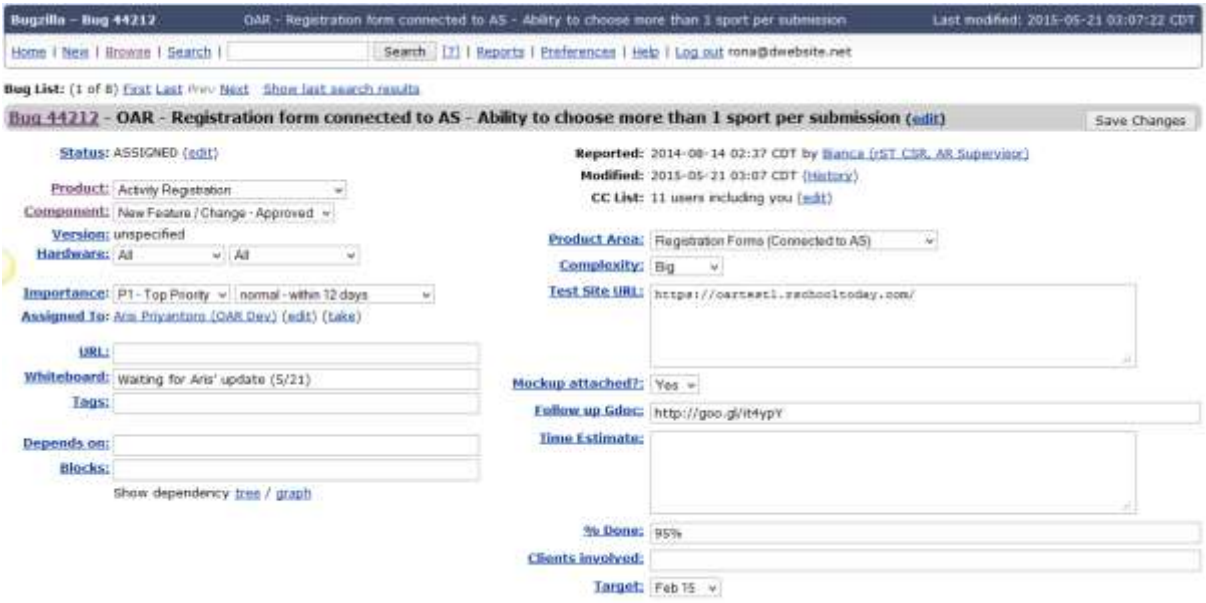


Gambar 2.1 Siklus atau *Workflow* Bugzilla

Keterangan

1. Confirmed / unconfirmed : status bug ticket dalam bugzilla yang tidak bisa dihapus atau diperbarui namanya.
2. In Progress : status bug ticket dalam Bugzilla di mana pengembang masih dalam proses pengerjaan tugas yang tercantum dalam deskripsi di Bugzilla.
3. Resolved : status dalam Bugzilla yang biasanya diubah oleh pihak tester / QA (Quality Assurance) apabila masalah yang ada sudah dites dan tidak lagi menemui kendala.
4. Verified : status dalam Bugzilla yang diubah oleh pihak tester apabila sudah tidak menemui kesalahan selama fase testing.

Contoh *Bug Ticket* Bugzilla



Gambar 2.3 Halaman Bug Ticket Bugzilla

Keterangan

1. Status : berisi status dari bug ticket
2. *Product* : nama produk yang dilaporkan
3. *Version* : biasanya berisi nama atau nomor dari versi produk yang telah dirilis yang berdampak pada bug yang dilaporkan
4. *Hardware* : perangkat keras yang digunakan saat bug ditemukan
5. *Importance* : berisi tingkat prioritas dan tingkat kepentingan dalam penyelesaian bug
6. *Assigned to* : perseorangan yang bertanggung jawab untuk memperbaiki bug
7. *URL* : URL yang berhubungan dengan bug bila ada
8. *Whiteboard* : untuk menambah catatan pendek terkait *bug*
9. *Tags* : berisi catatan acak yang berisi kata terkait penyelesain bug
10. *Depends on* : berisi catatatan acak keterkaitan bug ticket dengan *bug ticket* lain

11. *Product Area* : area produk yang di mana ditemukan bug

12. *Complexity* : tingkat kompleksitas *bug*
13 *Test Site URL* : berisi URL untuk fase tes
14. *Mockup attached* : berisi dokumen tambahan berupa *file* beragam format terkait dengan penyelesaian *bug*
15. *Follow up gdoc* : untuk menambah link dari gdoc terkait dengan *bug*
16. *Time Estimate* : estimasi waktu yang dibutuhkan untuk penyelesaian *bug*
17. *% Done* : persentase perbaikan *bug* yang sedang dilakukan pengembang
18. *Clients Involved* : pelanggan yang terlibat
19. Target : tanggal target penyelesaian dari *bug* yang sedang diperbaiki

2.3 Data Mining

Data mining adalah proses menemukan hubungan dalam data yang tidak diketahui oleh pengguna dan menyajikannya dengan cara yang dapat dipahami sehingga hubungan tersebut dapat menjadi dasar dalam pengambilan keputusan (McLeod & Schell, 2007).

Data mining memiliki enam tugas umum antara lain (Fayyad et.al, 2008)

1. Deteksi Anomali : identifikasi rekaman data yang tidak biasa, yang mungkin menarik atau data kesalahan yang memerlukan penyelidikan lebih lanjut
2. Aturan Asosiasi Belajar : pencarian untuk hubungan antar variabel
3. *Clustering* : tugas menemukan kelompok dan struktur dalam data yang dalam beberapa cara tanpa menggunakan struktur yang dikenal dalam data.

4. Klasifikasi : generalisasi struktur yang dikenal untuk diterapkan ke data baru.
5. Regresi : menemukan fungsi yang model data dengan meminimalisir kesalahan.
6. Penyimpulan : menyediakan representasi yang lebih kompak dari kumpulan data, termasuk visualisasi dan pembuatan laporan.

Pentingnya data mining saat ini terutama didorong oleh banyaknya data yang dikumpulkan dan disimpan dengan berbagai aplikasi terkini, seperti data web, data *e-commerce*, data pembelian, data keluhan, dan sebagainya. Data yang dihasilkan oleh aplikasi-aplikasi tersebut umumnya merupakan jenis *Big Data* dimana data tersebut sulit diolah atau dimengerti secara sederhana. *Big Data* merupakan data yang mempunyai tiga karakteristik yaitu jumlah (*volume*) dan variasi (*variety*) besar, serta bergerak cepat (*velocity*), sehingga melampaui kapasitas pengolahan *database* konvensional (Dumbill, 2014).

Penggunaan *data mining* dibedakan menjadi dua jenis fungsi yaitu prediktif dan deskriptif (Gullo, 2015). Penggalian prediktif mengacu pada pembangunan model yang berguna untuk memprediksi perilaku atau nilai-nilai di masa depan.

Tugas deskriptif meliputi klasifikasi dan prediksi, tugas yang dilakukan seperti membangun beberapa model (atau fungsi) yangmenggambarkan kelas atau konsep data oleh satu set objek data yang label kelasnya diketahui (*training set*), sehingga dapat memprediksi kelas yang labelnya tidak diketahui; deteksi penyimpangan, yaitu berurusan dengan penyimpangan data, yang didefinisikan sebagai perbedaan antara nilai yang terukur dan nilai referensi; analisis evolusi, yaitu, mendeteksi dan menggambarkan pola yang teratur dalam data yang perilakunya berubah dari waktu ke waktu.

Sedangkan tujuan penggalian deskriptif yaitu membangun model untuk mendeskripsikan data menjadi bentuk yang mudah dimengerti, efektif, dan efisien. Contoh dari tugas deskriptif diantaranya karakterisasi data, yang tujuan utamanya adalah untuk meringkas karakteristik umum atau fitur dari kelas target data; *association rule*, yaitu menemukan aturan yang menunjukkan kondisi

atribut-nilai yang sering muncul bersama-sama dalam himpunan data; dan *clustering*, yang bertujuan untuk membentuk kelompok yang memiliki kohesif tinggi dan terpisahkan dengan baik dari satu set objek data.

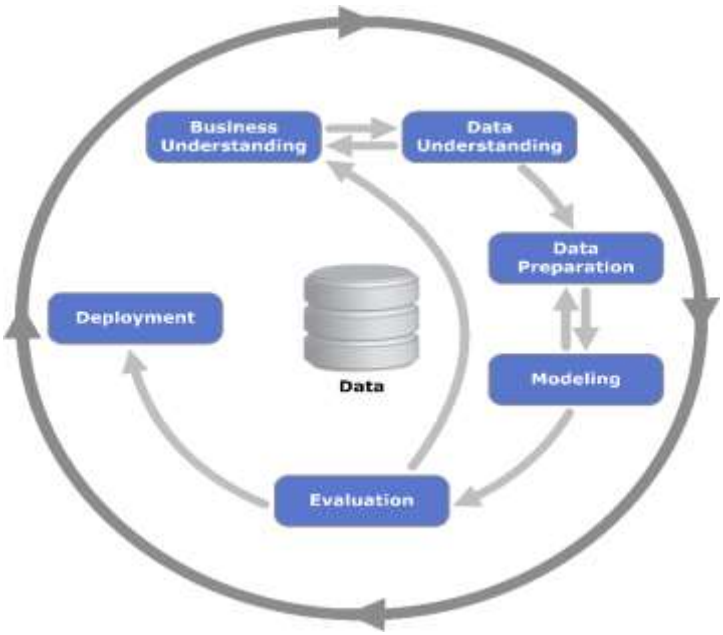
2.4 Metodologi CRISP-DM

CRISP-DM atau *Cross Industry Standard Process for Data Mining* adalah standardisasi *data mining* yang disusun oleh tiga penggagas *data mining market* yaitu Daimler Chrysler (Daimler-Benz), SPSS (ISL) dan NCR yang kemudian dikembangkan pada berbagai *workshops* antara tahun 1997-1999. Lebih dari 300 organisasi yang berkontribusi dalam proses modelling ini dan akhirnya CRISP-DM 1.0 dipublikasikan pada 1999 (Shearer, 2000).

Tidak ada ketentuan dan karakteristik tertentu untuk data yang dapat diproses dengan CRIPS-DM ini karena data diproses kembali dalam 6 fase di dalamnya. Fase-fase tersebut antara lain:

1. *Business Understanding* atau pemahaman penelitian. Pada fase ini dibutuhkan pemahaman tentang substansi dari kegiatan *data mining* yang akan dilakukan, kebutuhan dari perspektif bisnis. Kegiatannya antara lain: menentukan sasaran atau tujuan bisnis, memahami situasi bisnis, menentukan tujuan *data mining* dan menyiapkan strategi awal untuk tujuan.
2. *Data Understanding* atau pemahaman data adalah fase mengumpulkan data awal, mempelajari data untuk bisa mengenal data yang akan dipakai. Fase ini mencoba mengidentifikasikan masalah yang berkaitan dengan kualitas data, mendeteksi subset yang menarik dari data untuk membuat hipotesa awal. Jika data berasal dari lebih dari satu *database*, maka proses integrasi data berada pada proses ini.
3. *Data preparation* atau persiapan data. Fase ini sering disebut sebagai fase yang padat karya. Aktivitas yang dilakukan antara lain memilih kasus*dan* variabel yang akan dianalisis. Perubahan variabel juga bisa dilakukan apabila diperlukan
4. *Modeling* adalah fase menentukan teknik data mining yang digunakan, menentukan *tools data mining*, teknik data mining, algoritma data mining, menentukan parameter dengan nilai yang optimal.

5. *Evaluation* adalah fase interpretasi terhadap hasil data mining yang ditunjukan dalam proses pemodelan pada fase sebelumnya. Evaluasi dilakukan secara mendalam dengan tujuan menyesuaikan model yang didapat agar sesuai dengan sasaran yang ingin dicapai dalam fase pertama.
6. *Deployment* atau penyebaran adalah fase penyusunan laporan atau presentasi dari pengetahuan yang didapat dari evaluasi pada proses *data mining*.



Gambar 2.4 Proses CRISP-DM

2.5 Text Mining

Text mining adalah salah satu bidang khusus dalam *data mining* yang memiliki definisi menambang data berupa teks dimana sumber data biasanya didapatkan dari dokumen dan tujuannya adalah mencari kata-kata yang dapat mewakili isi dari dokumen sehingga dapat dilakukan analisa keterhubungan antar dokumen (Mooney, 2006).

Text mining dapat menganalisa dokumen, mengelompokkan dokumen berdasarkan kata-kata yang terkandung di dalamnya, serta menentukan kesamaan di antara dokumen untuk mengetahui bagaimana mereka berhubungan dengan variabel lainnya (Statsoft, 2015). Penerapan yang paling umum dilakukan text mining saat ini misalnya penyaringan spam, analisa sentimen, mengukur preferensi pelanggan, meringkas dokumen, pengelompokan topik penelitian, dan banyak lainnya.

Text Mining sendiri memiliki beberapa tipe antara lain (Abbot, 2013) :

1. *Search and Information Retrieval*

   Menyimpan dan menemukan kembali dokumen teks, termasuk mesin pencari dan kata kunci pencarian.

2. *Document Clustering*

   Pengelompokan dan pengkategorian istilah, potongan, paragraf, atau dokumen menggunakan metode mining .

3. *Document Classification*

   Pengelompokan dan pengkategorian istilah, potongan, paragraf, atau dokumen menggunakan metode *document classification*.

4. *Web Mining*

   Data dan text mining pada internet yang fokus pada skala dan antar hubungan pada website.

5. *Information Extraction*

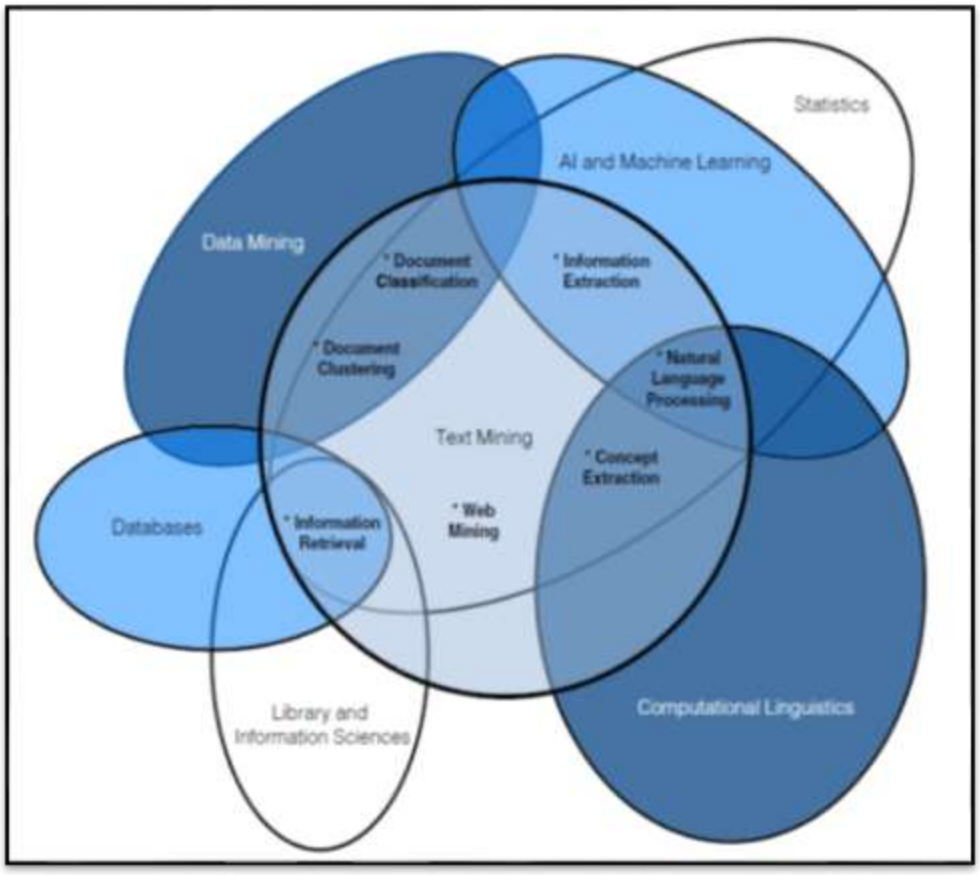
   Identifikasi dan ekstraksi fakta yang relevan.

6. *Natural Language Processing*

   Pemrosesan bahasa tingkat rendah yang biasanya digunakan untuk bahasa komputasi.

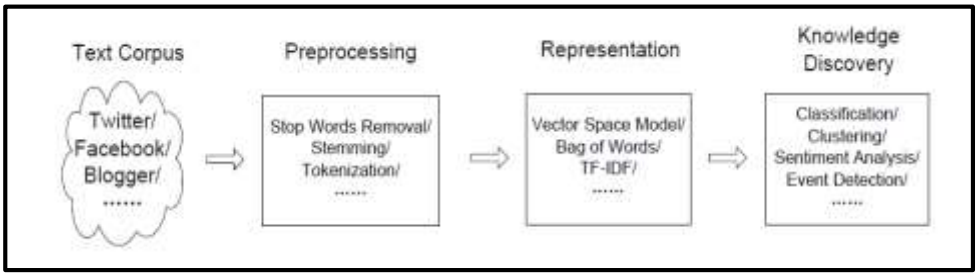
7. *Concept Extraction*

   Pengelompokan kata dan frase dalam grup yang sama.



Gambar 2.5 Diagram venn 6 bidang terkait dan 7 area praktek *text mining*

Untuk memperoleh tujuan seperti ditunjukkan pada Gambar 2.6. Data terpilih yang akan dianalisis pertama akan melewati tahap Pra-proses dan representasi teks, hingga akhirnya dapat dilakukan *knowledge discovery.*



Gambar 2.6 Kerangka proses analisis teks pada *text mining*

Tahapan pra-pemrosesan dalam *Text Mining* antara lain menurut Mooney (2006) terdiri dari beberapa fitur antara lain:

1. *Tokenizing* : tahap pemotongan string input berdasarkan tiap kata yang menyusunnya

   Contoh dari tahap ini adalah sebagai berikut:

   Tabel 2.1 Tahap Tokenizing

Teks Input	Hasil Token
New London Spicer suggested that the system tracks	New London Spicer suggested that the system track

2. *Filtering* : tahap pengambilan kata-kata yang penting dari hasil token. Bisa menggunakan algoritma *stop list* (membuang kata yang kurang penting) atau *word list* (menyimpan kata yang penting)

   Tabel 2.2 Tahap Filtering

Hasil Token	Hasil Filter
New London Spicer suggested that the system tracks	New London Spicer suggested that system tracks

3. *Stemming* : tahap mencari akar kata hasil filtering

   Tabel 2.3 Tahap Stemming

Hasil Token	Hasil Stemming
Learning using text mining	Learn use text

	mine

4. *Tagging* : tahap mencari bentuk awal dari kata lampau hasil stemming

   Tabel 2.4 Tahap Tagging

Hasil Stemming	Hasil tagging
Was Used Story	Be Use Story

5. *Analyzing* : tahap penentuan seberapa jauh keterhubungan antar kata-kata antar dokumen yang ada.

2.5.1 Metode N-gram

N-gram adalah potongan sejumlah n karakter dari sebuah string (Furnkraz, 2009). Ngram merupakan sebuah metode yang diaplikasikan untuk pembangkitan kata atau karakter. Metode n-gram ini digunakan untuk mengambil potongan-potongan karakter huruf sejumlah n dari sebuah kata yang secara kontinuitas dibaca dari teks sumber hingga akhir dari dokumen.

N-gram dibedakan berdasarkan jumlah potongan karakter sebesar n. Untuk membantu dalam mengambil potongan-potongan kata berupa karakter huruf tersebut, maka dilakukan *padding* dengan blank di awal dan di akhir suatu kata. Sebagai contoh : kata ”TEXT” dapat diuraikan ke dalam beberapa n-gram berikut (”_” merepresentasikan blank): uni-grams : T, E, X, T

bi-grams : _T, TE, EX, XT, T_

tri-grams : _TE, TEX, EXT, XT_

quad-grams : _TEX, TEXT, EXT_

quint-grams : _TEXT, TEXT_

Salah-satu keunggulan menggunakan N-gram dan bukan suatu kata utuh secara keseluruhan adalah bahwa N-gram tidak akan terlalu sensitif terhadap kesalahan penulisan yang terdapat pada suatu dokumen (Furnkraz, 2009).

2.5.2 Pembobotan TF-IDF

Dalam rangka membangun model vektor, perlu dilakukan proses pembobotan. Skema pembobotan yang paling banyak digunakan adalah skema *term frequency-inverse document frequency* (TF-IDF).Pembobotan TF-IDF ini digunakan karena efisien, mudah, dan memiliki hasil yang akurat (Robertson, 2004). *Term frequency* (TF) didefinisikan sebagai jumlah kemunculan suatu kata/istilah dalam suatu dokumen. Misalnya TF pada dokumen pertama untuk kata/istilah "*merge*" adalah 2, karena kata/istilah tersebut muncul 2 kali dalam dokumen pertama.

Pada asumsi pembobotan dibalik TF-IDF, kata-kata dengan nilai TF yang tinggi akan mendapat bobot yang tinggi kecuali jika jumlah dokumen yang mengandung kata tersebut juga tinggi yang disebut *inverse document frequency*atau IDF. Misalnya kata "*please*" memiliki jumlah kemunculan yang tinggi tetapi jumlah dokumen yang mengandung kata "*please*" juga tinggi, sehingga kata tersebut akan memiliki bobot yang rendah.

Skema persamaan TF – IDF ditunjukkan oleh persamaan seperti di bawah ini (Zhai & Aggarwal, 2012).

$$\text{tf idf (w)} = \text{tf x log}\frac{N}{df\ (w)} \tag{2.1}$$

Keterangan:

1. tf (w) = *term frequency* (jumlah kemunculan suatu kata dalam dokumen)
2. df (w) = *document frequency* (jumlah dokumen yang mengandung suatu kata)
3. N = jumlah dokumen

2.6 Ekstraksi Informasi pada Text Mining

Tahap akhir penggalian informasi pada text mining yaitu ektraksi ilmu pengetahuan (*knowledge discovery*), dimana terdapat beberapa jenis kelas utama yang bisa dilakukan sebagai berikut (Miner et al, 2012).

1. Klasifikasi/prediksi,

Klasifikasi adalah bentuk analisis data yang mengekstrak model untuk menggambarkan kelas data (Jiawei, Kamber, & Pei, 2012). Model yang dibangun meliputi pengklasifikasian dan prediksi label kelas. Klasifikasi data mempunyai dua tahapan proses, yaitu tahap pembelajaran (learning step) dimana model klasifikasi dibangun berdasarkan label yang sudah diketahui sebelumnya dan tahapan klasifikasi (*classification step*) dimana model digunakan untuk memprediksi label kelas dari data yang diberikan (Miner et al, 2012).

2. Pengelompokan (*clustering*)

Tidak seperti klasifikasi, kelompok label kelas pada model clustering tidak diketahui sebelumnya dan tugas clustering adalah untuk mengelompokkannya (Linoff & Berry, 2011). Menurut Linof & Berry (2011), *clustering* adalah proses pengelompokan satu set data objek menjadi beberapa kelompok atau klaster sehingga objek dalam sebuah klaster memiliki kemiripan yang tinggi satu sama lain, tetapi sangat berbeda dengan objek dalam kelompok lainnya.

3. Asosiasi

Asosiasi merupakan proses pencarian hubungan antar elemen data. Dalam dunia industri retail, analisis asosiasi biasanya disebut *Market Basket Analysis* (Miner et al, 2012). Asosiasi tersebut dihitung berdasarkan ukuran support (presentase dokumen yang memuat seluruh konsep suatu produk A dan B) dan confidence (presentase dokumen yang memuat seluruh konsep produk B yang berada dalam subset yang sama dengan dokumen yang memuat seluruh konsep produk A).

4. Analisis Tren

Tujuan dari analisis tren yaitu untuk mencari perubahan suatu objek atau kejadian oleh waktu (Miner, et al 2012). Salah satu aplikasi analisis tren yaitu kegiatan identifikasi evolusi topik penelitian pada artikel akademis.

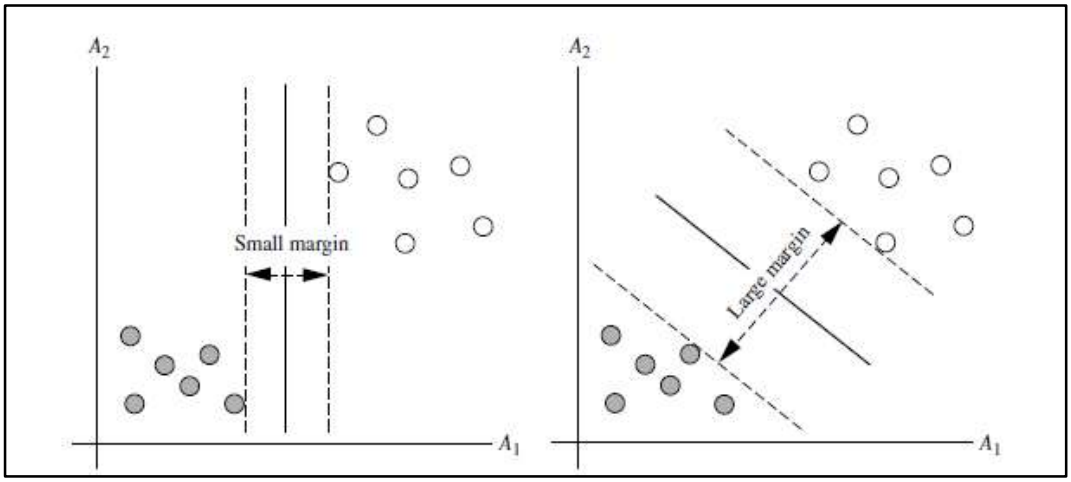
2.7 Algoritma Penggolongan Klasifikasi

Pada studi klasifikasi, proses pembelajaran dilakukan berdasarkan prinsip *machine learning*. *Machine learning* merupakan suatu metode yang menyelidiki bagaimana komputer belajar mengenai data (Jiawei, Kamber, & Pei, 2012). Dalam *machine learning*, *training model* (model latihan) akan dipelajari dengan

menggunakan berbagai algoritma yang ditentukan untuk mendapatkan model pengklasifikasi yang dapat digunakan untuk mengklasifikasikan dokumen lainnya yang belum mempunyai kelas sebelumnya. Algoritma yang biasanya digunakan untuk melakukan pengklasifikasian antara lain adalah *Support Vector Machine* (SVM) dan Naïve Bayes Classifier.

2.8 Support Vector Machine

SVM merupakan algoritma klasifikasi yang memiliki tujuan untuk menemukan fungsi pemisah (*hyperplane*) dengan margin paling besar, sehingga dapat memisahkan dua kumpulan data secara optimal (Jiawei, Kamber, & Pei, 2012). Gambar 2.7 menunjukkan dua *hyperplane* yang mungkin untuk memisahkan dua kelompok data. Kedua *hyperplane* dapat mengklasifikasikan semua tupel data yang diberikan, tetapi *hyperplane* dengan margin yang lebih besar mempunyai tingkat akurasi lebih tinggi dalam melakukan klasifikasi karena dapat memisahkan kumpulan data yang satu dengan lainnya dengan mencari tingkat pemisah yang paling jauh antar kelompok.



Gambar 2.7 *Margin Hyperplane* SVM

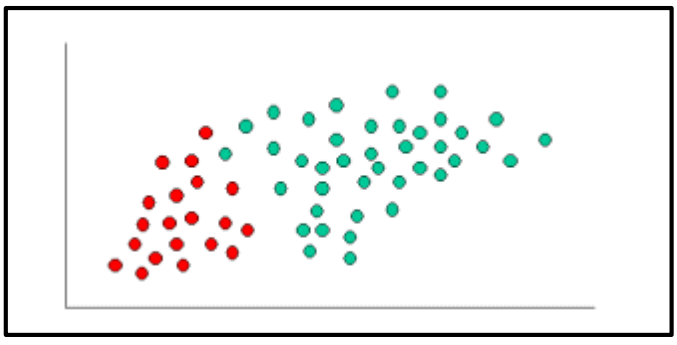
SVM pada awalnya digunakan untuk klasifikasi data numerik, tetapi ternyata SVM juga sangat efektif dan cepat untuk menyelesaikan masalah-masalah data teks. Data teks cocok untuk dilakukan klasifikasi dengan algoritma SVM karena sifat dasar teks yang cenderung mempunyai dimensi yang tinggi, dimana terdapat beberapa fitur yang tidak relevan, tetapi akan cenderung

berkorelasi satu sama lain dan umumnya akan disusun dalam kategori yang terpisah secara linear (Zhai & Aggarwal, 2012).

2.8.1 Naive Bayes Classifier

Naïve Bayes Classifier merupakan pengklasifikasi probabilistik sederhana yang didasarkan pada teorema Bayes, yang menyatakan bahwa kemungkinan terjadinya suatu peristiwa sama dengan probabilitas intrinsik (dihitung dari data yang tersedia sekarang) dikalikan probabilitas bahwa hal serupa akan terjadi lagi di masa depan (berdasarkan pengetahuan yang terjadinya di masa lalu) (Miner et al, 2012). PengklasifikasianNaïve Bayes memiliki asumsi bahwa efek dari suatu nilai atribut tertentu tidak bergantung (independen) terhadat nilai atribut lainnya (Zhai & Aggarwal, 2012). Asumsi tersebut disebut *class conditional indepence*.

Pengklasifikasi ini dapat sangat efisien dan akurat, terutama ketika jumlah variabel yang tinggi. Contoh sederhana dalam perhitungan *Naive Bayes*misalnya terlihat padaGambar 2.8 terdapat dua kumpulan data yaitu hijau dan merah (Statsoft, 2015). Data baru akan ditambahkan dan akan ditentukan data baru tersebut merupakan bagian dari kelas yang mana. Karena jumlah data hijau dua kali lebih banyak daripada merah, maka diasumsikan bahwa data yang baru memiliki probabilitas menjadi anggota hijau dua kali lebih besar dari merah. Dalam analisa Bayesian, keyakinan ini dikenal sebagai probabilitas prior. Probabilitas prior didasarkan pada pengalaman sebelumnya, dalam hal ini persentase data hijau dan merah.



Gambar 2.8 Dua kelompok data *Naïve Bayes*

$$\text{Probabilitas prior untuk hijau} = \frac{Jumlah\ data\ hijau}{Jumlah\ data\ \boldsymbol{keseluruhan}} \qquad (2.2)$$

$$\text{Probabilitas prior untuk merah} = \frac{Jumlah\ data\ merah}{Jumlah\ data\ \boldsymbol{keseluruhan}} \quad (2.3)$$

2.9 Penelitian Terdahulu

Dari berbagai telaah tentang keluhan pelanggan dan data mining, maka ditemukan beberapa jurnal dan penelitian sebagai berikut:

Tabel 2.5 Penelitian Terdahulu

Judul, Peneliti, dan Tahun Publikasi	**Obyek Penelitian**	**Teknik yang Digunakan**
Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews (Dave ,Lawrence,Pennock 2003)	Produk Cnet dan Amazon	Support Vector Machine dan Naive Bayessian
Churn prediction in subscription services: Anapplication of support vector machines while comparing two parameter selection techniques (Coussement & Van den Poel, 2006	Email dari Call Center	Frekuensi istilah
Extracting Aspect-Evaluation and Aspect-of Relations in Opinion Mining (Kobayashi,Inui,Matsumoto, 2007)	Restoran dan seluler	Klasifikasi dan aspek antar hubungan
Red Opal: Product-Feature Scoring from Reviews (Scaffidi et.al, 2007)	Amazon	Pola dan frekuensi istilah, klasifikasi
A New Approach of Using	Pengaduan pada	CRM dan algoritma Apriori

Association Rule Mining in Customer Complaint Management (Bigdoli &Akhondzadeh, 2010)	kantor pemerintahan	
Clustering Product Features for Opinion Mining (Zhai et.al, 2011)	Review Produk	Naive Bayessian, clustering
Analyzing Costumer Experience Feedback Using Text Mining: A Linguistic - Based (Ordenes et.al, 2014)	Parkir bandara	Framework ARC

2.10 QDA Miner

QDA Miner adalah perangkat lunak yang dikembangkan oleh Provalis Research untuk menganalisa data dengan metode campuran dan analisa data kualitatif. Perangkat lunak ini pertama kali dirilis pada tahun 2004 setelah pengembangannya diselesaikan oleh Norman Peladeau. Versi terakhir dari QDA Miner yaitu versi 4 dirilis pada tahun Desember 2011 dan digunakan untuk penelitian penjualan, survey perusahaan, pemerintah, penelitan dalam bidang pendidikan, penelitian kriminalitas, jurnalistik, dan masih banyak lagi (Provalis Research, 2012).

Fitur dalam QDA Miner antara lain:

1. Impor data dalam beragam jenis format dokumen dan gambar antara lain: PDF, Word, Excel, HTML, RTF, SPSS, JPEG
2. *Tools* Temu Balik : Keyword Retrieval, Query-by-Example, Cluster Extraction
3. Statistika : *Coding frequencies, cluster analysis, coding sequences, coding by variables.*
4. *Tools* Visualisi : multidimensional scaling, heatmaps, grafik koresponden, proximity plot.
5. GeoTagging (GIS) dan Time-Tagging tools
6. Tools untuk manager pelaporan yang menyimpan kueri dan hasil analisa,

tabel, grafik dan catatan penelitian.

2.11 Studi Pustaka

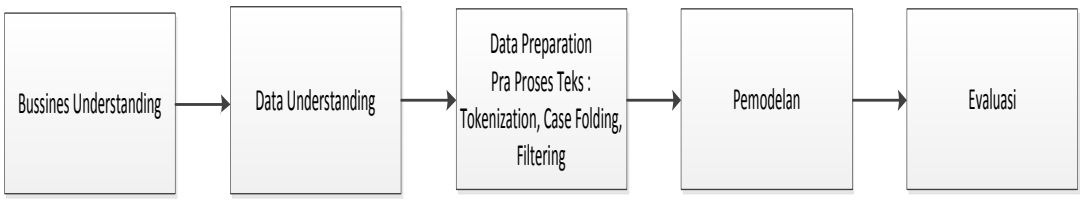
Studi pustaka dilakukan dengan cara mengumpulkan serta mempelajari informasi-informasi yang didapat dari buku, artikel, situs, serta sumber bacaan lain yang memiliki keterkaitan dengan permasalahan yang akan diselesaikan pada penelitian ini. Studi pustaka pada penelitian ini menggunakan teori-teori metode *Cross Industry Standard Process for Data Mining* (CRISP–DM), *text mining*,algoritma *SVM* dan *Naive Bayes Classification*, pembobotan *TF IDF* dan *bug tracking analysis*. Selain informasi yang didapat dalam buku, penelitian ini juga menggunakan artikel atau *paper* yang didapat dari media *online*.

BAB III

Metodologi Penelitian

3.1 Langkah – langkah penelitian

Dalam penelitan ini digunakan langkah-langkah dalam menyelesaikan permasalahan. Secara garis besar penelitian ini menggunakan langkah-langkah sebagai berikut



Gambar 3.1 Garis Besar Langkah – langkah Penelitian

3.2 Metode CRISP-DM

Metodologi *Cross Industry Standard Process for Data Mining* (CRISP–DM) digunakan sebagai standard proses *data mining*yang dalam fase model disertakan teknik *text mining*sekaligus sebagai metode penelitian.

3.2.1 Bussiness Understanding atau Pemahaman Bisnis

Tahap pertama dalam proses CRISP-DM bisa juga disebut tahap pemahaman penelitian. Tahap ini digunakan untuk memahami manfaat dari kegiatan.

3.2.2 Data Understanding atau Pemahaman Data

Pada tahap ini mengumpulkan data awal terlebih dahulu sehingga bisa dipahami dan data-data tersebut bisa dikenali sehingga bisa ditarik kesimpulan apa saja yang bisa dilakukan oleh data-data tersebut.

Pada penelitian ini data didapat dari *database* Bugzilla perusahaan pada proyek *Activity Registration* (AR). Data diambil dari *database* mulai bulan Juni 2014 (awal pembuatan sistem) hingga September 2015 (setelah dirilis ke beberapa klien). Data yang digunakan dalam penelitian ini terdiri dari empat kelas yaitu *Registration Form*, *Setup*, *Connection to AS,* dan *Family Account*. Sebanyak 240 data keluhan dari customer service digunakan dalam penelitian ini. Data keluhan kemudian dibagi menjadi 2 yaitu 200 data *training* dan 40 data *testing*. Dari ke - 200 data training, masing-masing kelas keluhan memiliki jumlah data yang sama yaitu 50 data. Apabila model klasifikasi telah dibuat, kemudian akan diuji dengan 40 data testing dimana masing-masing kategori kelas terdiri dari 10 data *testing*.

3.2.3 Data Preparation atau Persiapan Data

Data preparation atau persiapan dokumen meliputi pra-proses teks,*converting, tokenization, case folding,* dan *filtering*. *Converting* adalah proses mengubah dokumen yang awalnya berupa *file* sql menjadi dokumen berekstensi excel. *Filtering* adalah proses menghilangkan *stopwords* dan tanda baca.

3.2.3.1 Pra Proses Teks

Keluhan yang bersifat tekstual yang bersifat tidak terstruktur menjadi model yang terstruktur. Model yang terstruktur diperlukan agar data bisa diolah dan dianalisa dengan menggunakan *text mining*. Pra-proses teks terdiri dari berberapa tahap, dimana setiap tahap dapat dilakukan secara manual. Akan tetapi, dikarenakan data laporan mempunyai jumlah yang sangat besar dan memerlukan waktu yang cukup banyak untuk melakukan seluruh tahap pra-proses secara manual, maka digunakan aplikasi QDA Miner agar pra-pemrosesan berjalan secara singkat, tepat dan sesuai dengan kebutuhan

a. *Tokenization*

Tokenization merupakan proses pemisahan teks menjadi potongan kata yang disebut *token. Tokenization* dilakukan untuk mendapatkan token atau potongan kata yang akan menjadi entitas yang memiliki nilai dalam penyusunan matriks dokumen pada proses selanjutnya.

b. *Case Folding*

Case folding merupakan proses pengubahan huruf dalam dokumen menjadi satu bentuk, misalnya huruf kapital menjadi huruf kecil dan sebaliknya

c. *Filtering*

Proses persiapan dokumen selanjutnya adalah *filtering*, yaitu menghilangkan *stopwords* dan tanda baca. *Stopwords* adalah daftar kata-kata yang tidak dipakai di dalam pemrosesan bahasa alami (kata depan, kata penghubung,kata pengganti). Keseluruhan daftar *stopwords* sudah diolah dalam QDA Miner. Selain menghilangkan *stopwords* dan tanda baca, proses filtering juga menghilangkan karakter ASCII 0 hingga 31 yang belum hilang setelah proses converting dokumen. Karakter-karakter ASCII 0 hingga 31 dapat menganggu proses pembacaan pada saat klasifikasi dokumen.

3.2.4 Pemodelan

Pemodelan adalah fase yang melibatkan pembobotan. Dalam penelitian ini teknik yang digunakan adalah fitur N-gram, algoritma *Support Vector Machine* dan *Naive Bayes Classification* dan TF IDF.

1. Fitur N Gram: Fitur n-gram digunakan dalam proses pembuatan model dengan membagi suatu kalimat menjadi beberapa bagian kata. Dalam penelitian ini, dilakukan perbandingan tiga buah fitur n-gram yaitu unigram, bigram, dan trigram. Dalam n-gram, 'n' menunjukkan jumlah kata yang akan dikelompokkan menjadi satu bagian. Misalnya, apabila n=2 atau biasa disebut dengan bigram, maka sebuah kalimat akan dipecah menjadi masing-masing dua kata pada setiap bagian. Apabila terdapat kalimat *"Please merge this account"*, maka dengan fitur bigram akan dipecah menjadi sebagai berikut:

   Bagian 1 : *please merge*

   Bagian 2 : *merge this*

   Bagian 3 : *this account*
2. *Term Frequency*: *Term frequency* merupakan salah satu metode yang digunakan untuk melakukan perhitungan pembobotan *term*. Fitur *term* frequency dilakukan dengan menghitung frekuensi kemunculan *term* tertentu pada suatu dokumen.
3. IDF: Inverse Document Frequency berfungsi mengurangi bobot *term* jika kemunculannya banyak menyebar di dokumen.

3.2.5 Evaluasi

Evaluasi adalah fase interpretasi terhadap hasil *text mining*. Evaluasi dilakukan secara mendalam dengan tujuan agar hasil pada tahap permodelan sesuai dengan sasaran yang ingin dicapai dalam tahap *business understanding*.

Model klasifikasi yang dibangun perlu dievaluasi untuk mengetahui seberapa baik model tersebut dalam melakukan klasifikasi yang diinginkan. Dalam mengevaluasi kinerja pengklasifikasi khususnya klasifikasi teks umumnya dilakukan dengan akurasi atau dengan *precision* and *recall* (Miner, et al, 2012). Nilai akurasi merepresentasikan seberapa banyak keseluruhan dokumen diklasifiksikan dengan benar.

Pada kasus ketidakseimbangan kelas/kategori pada data latihan dimana terdapat kelas data mayoritas dan minoritas, seringkali nilai akurasi kurang bisa merepresentasikan performa model secara signifikan (Jiawei, Kamber, & Pei, 2012). Nilai akurasi memberikan nilai 97%, dengan nilai ini model klasifikasi bisa dikatakan sudah sangat akurat. Namun, bisa saja 97% tersebut hanya mendeteksi secara benar, dan 3% sisanya salah mendeteksi. Untuk mengatasinya, pengukuran *precision and recall* biasa dilakukan dalam mengevaluasi model klasifikasi. Selain mampu menunjukkan keakuratan model secara keseluruhan, pengukuran ini juga mampu menunjukkan bagaimana performa model pada setiap kelas.

Pengukuran *precision* dan *recall* merupakan matrik evaluasi yang paling sering digunakan pada kasus klasifikasi teks (Sokolova & Lapalme, 2009). Misalnya terdapat dua kelas A dan B, *precision* yaitu jumlah sampel berkelas A yang ditebak dengan benar sebagai A dibanding dengan jumlah total data yang ditebak sebagai A, sedangkan *recall* yaitu jumlah sampel berkelas A yang ditebak dengan benar dibandingkan dengan jumlah total sampel A.

Pada penelitian ini mekanisme yang dapat digunakan untuk mengukur validitas hasil klasifikasi adalah dengan menghitung nilai *precision, recall*, dan *f-score*. Perhitungan nilai *precision* akan mengukur tingkat kepastian (*exactness*) atau jumlah data *testing* yang diklasifikasikan dengan benar oleh model klasifikasi yang dibangun. Perhitungan *recall* merupakan kebalikan dari *precision. Recall* mengukur sensitifitas atau rasio dari data untuk setiap label yang diklasifikasikan dengan benar terhadap data yang salah diklasifikasikan

ke label lainnya (*missclassified*). F-score merupakan *trade-off* antara *precision* dan *recall*. Nilai f-score didapat dengan menghitung *harmonic mean* antara *precision* dan *recall*.

Menurut Lancaster (1979) perhitungan *recall* dan *precision* dengan rumus sebagai berikut:

$$Recall = \frac{\text{Jumlah dokumen relevan yang terpanggil (a)}}{\text{Jumlah dokumen relevan yang ada di dalam } database \text{ (a+c)}} \text{x}100 \qquad (3.1)$$

$$Precision = \frac{\text{Jumlah dokumen relevan yang terpanggil (a)}}{\text{Jumlah dokumen relevan yang ada di dalam pencarian (a+b)}} \text{x}100 \qquad (3.2)$$

Tabel 3.1 Perhitungan *Recall* dan *Precission*

	Relevant	**Not Relevant**	**Total**
Retrieved	a (hits)	b (noise)	a+b
Not Retrieved	c (misses)	d (reject)	c+d
Total	a+c	b+d	a+b+ c+d

Keterangan:

1. a (hits) = dokumen yang relevan
2. b (noice) = dokumen yang tidak relevan
3. c (misses) = dokumen relevan yang tidak ditemukan
4. d (reject) = dokumen yang tidak relevan yang tidak ditemukan

F-measure atau F_1 merupakan salah satu perhitungan evaluasi dalam temu kembali informasi yang mengkombinasikan *recall* dan *precision*. Nilai *recall* dan *precision* pada suatu keadaan dapat memiliki bobot yang berbeda. Ukuran yang menampilkan timbal balik antara *recall* dan *precision* adalah F-measure yang merupakan bobot *harmonic mean* dari *recall* dan *precision*.

Menurut Manning (Manning, 2009), memisahkan dokumen-dokumen yang mirip kadang lebih buruk daripada menempatkan pasangan dokumen yang tidak mirip ke dalam *cluster* yang sama. Dengan demikian, dapat digunakan F-Measure dengan nilai *false negative* lebih kuat dari nilai *false positive*. Selanjutnya, akan

diberikan nilai $\beta > 1$ sehingga memberikan bobot yang lebih untuk *recall*. F-Measure yang seimbang memberikan bobot yang sama antara *recall* dan *precision*, dengan nilai $\alpha = ½$ atau $\beta = 1$. Range dari nilai F-Measure adalah 0 sampai dengan 1. Berikut ini adalah rumus dari F-Measure atau F_1

$$\mathrm{F1} = 2.\frac{precission.recall}{precission + recall} \tag{3.3}$$

Bab IV
Hasil dan Pembahasan

4.1 Bussiness Understanding atau Pemahaman Bisnis

Sasaran dari penelitian ini adalah aplikasi Activity Registration. dari tahap ini antara lain aplikasi yang menjadi sasaran adalah Activity Registration dan bertujuan untuk menggali keluhan dari para klien.

4.2 Data Understanding atau Pemahaman Data

Data diambil dari *database* mulai bulan Juni 2014 (awal pembuatan sistem) hingga September 2015 (setelah dirilis ke beberapa klien).

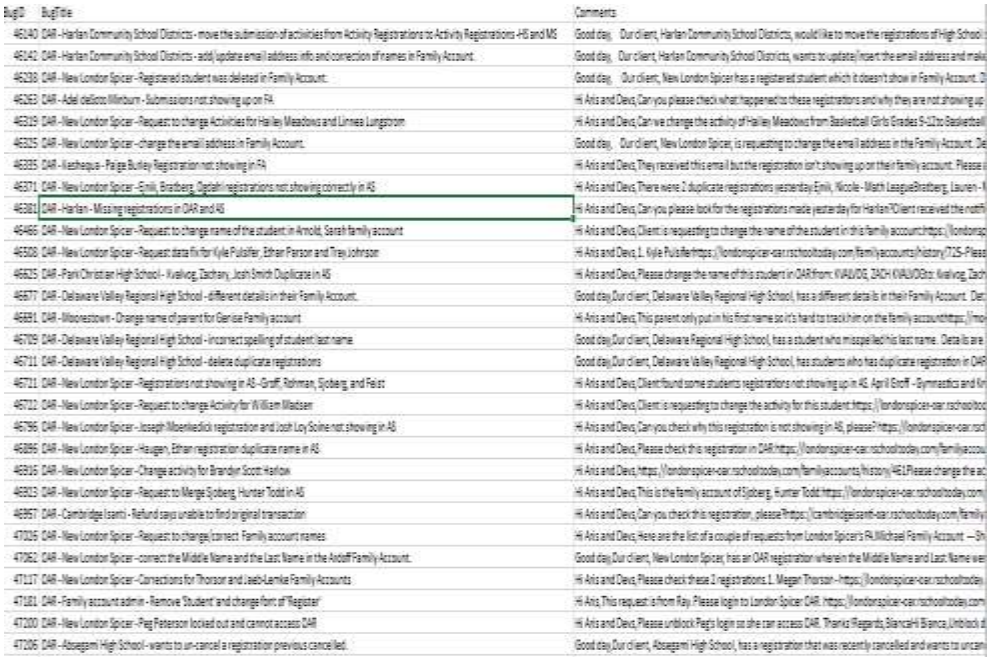
Tabel 4.1 Dataset

Class Label	**Data Training**	**Data Testing**
Registration Form	50	10
Setup	50	10
Connection to AS	50	10
Family Account	50	10
Total	200	40

4.3 Data Preparation atau Persiapan Data

Pemahaman data mengacu pada data yang terdapat pada *database* Bugzilla untuk produk AR. Tahap ini memahami format data secara permukaan dan lebih mendalam sekaligus menentukan mana saja yang layak untuk dijadikan atribut untuk mengklasifikasikan ragam keluhan pengguna perangkat lunak.

Dalam tahap ini, penggunaan teknik *text mining* digunakan pada kolom *comments* karena sangat banyak karakter berupa slash (/), dash (-), titik koma (;), titik (.), tanda petik (“) yang tidak diperlukan untuk menganalisa ragam keluhan. Hanya diperlukan beberapa kata untuk mengetahui keluhan apa saja yang ada dalam satu kolom c*omments.*



BugID	BugTitle	Comments
46140	OAR - Harlan Community School Districts - move the submission of activities from Activity Registrations to Activity Registrations -HS and MS	Good day, Our client, Harlan Community School Districts, would like to move the registrations of High School
46142	OAR - Harlan Community School Districts - add/update email address info and connection of names in Family Account.	Good day, Our client, Harlan Community School Districts, wants to update/insert the email address and make
46238	OAR - New London Spicer - Registered student was deleted in Family Account.	Good day, Our client, New London Spicer has a registered student which it doesn't show in Family Account. D
46263	OAR - Adel dalSoo Winburn - Submissions not showing up on PA	Hi Aris and Devs, Can you please check what happened to these registrations and why they are not showing up
46319	OAR - New London Spicer - Request to change Activities for Hailey Meadows and Linnea Lungstrom	Hi Aris and Devs, Can we change the activity of Hailey Meadows from Basketball Girls Grades 5-12 to Basketball
46335	OAR - New London Spicer - change the email address in Family Account.	Good day, Our client, New London Spicer, is requesting to change the email address in the Family Account. De
46335	OAR - Kashequa - Paige Burley Registration not showing in PA	Hi Aris and Devs, They received this email but the registration isn't showing up on their family account. Please
46371	OAR - New London Spicer - Emili, Bradberg, Digdahl registrations not showing correctly in AS	Hi Aris and Devs, There were 2 duplicate registrations yesterday Emili, Nicole - Math LeagueBradberg, Lauren - M
46381	OAR - Harlan - Missing registrations in OAR and AS	Hi Aris and Devs, Can you please look for the registrations made yesterday for Harlan?Client received the notifi
46486	OAR - New London Spicer - Request to change name of the student in Arnold, Sarah family account	Hi Aris and Devs, Client is requesting to change the name of the student in this family account:https://londonsp
46508	OAR - New London Spicer - Request data fix for Kyle Pulcifer, Ethan Parson and Trey Johnson	Hi Aris and Devs, 1. Kyle Pulcifer:https://londonspicer-oar.rschooltoday.com/family/accounts/history/7215-Please
46635	OAR - Paris Christian High School - Kvalvog, Zachary, Josh Smith Duplicate in AS	Hi Aris and Devs, Please change the name of this student in OAR from: KVAUDG, ZACH KVAUDGto: Kvalvog, Zach
46677	OAR - Delaware Valley Regional High School - different details in their Family Account.	Good day,Our client, Delaware Valley Regional High School, has a different details in their Family Account. Det
46691	OAR - Moorestown - Change name of parent for Gerise Family account	Hi Aris and Devs, This parent only put in his first name so it's hard to track him on the family account:https://mo
46709	OAR - Delaware Valley Regional High School - incorrect spelling of student last name	Good day,Our client, Delaware Regional High School, has a student who misspelled his last name. Details are
46711	OAR - Delaware Valley Regional High School - delete duplicate registrations	Good day,Our client, Delaware Valley Regional High School, has students who has duplicate registration in OAR
46721	OAR - New London Spicer - Registrations not showing in AS - Graff, Rahman, Sjoberg, and Feist	Hi Aris and Devs, Client found some students registrations not showing up in AS. April Graff - Gymnastics and Kr
46722	OAR - New London Spicer - Request to change Activity for William Madsen	Hi Aris and Devs, Client is requesting to change the activity for this student:https://londonspicer-oar.rschoolto
46796	OAR - New London Spicer - Joseph Moenkedick registration and Josh Loy Soine not showing in AS	Hi Aris and Devs, Can you check why this registration is not showing in AS, please? https://londonspicer-oar.rsch
46896	OAR - New London Spicer - Haugen, Ethan registration duplicate name in AS	Hi Aris and Devs, Please check this registration in OAR:https://londonspicer-oar.rschooltoday.com/family/accou
46916	OAR - New London Spicer - Change activity for Brandon Scott-Harlow	Hi Aris and Devs, https://londonspicer-oar.rschooltoday.com/family/accounts/history/461/Please change the ac
46933	OAR - New London Spicer - Request to Merge Sjoberg, Hunter Todd in AS	Hi Aris and Devs, This is the family account of Sjoberg, Hunter Todd: https://londonspicer-oar.rschooltoday.com/
46957	OAR - Cambridge (sent) - Refund says unable to find original transaction	Hi Aris and Devs, Can you check this registration, pleasehttps://cambridgesent-oar.rschooltoday.com/family
47026	OAR - New London Spicer - Request to change/correct Family account names	Hi Aris and Devs, Here are the list of a couple of requests from London Spicer's Ri.Wilchael Family Account —Sh
47062	OAR - New London Spicer - correct the Middle Name and the Last Name in the Ardoff Family Account.	Good day,Our client, New London Spicer, has an OAR registration where in the Middle Name and Last Name wer
47117	OAR - New London Spicer - Corrections for Thorson and Laebl-Lemke Family Accounts	Hi Aris and Devs, Please check these 2 registrations.1. Megan Thorson - https://londonspicer-oar.rschooltoday.
47181	OAR - Family account admin - Remove 'Student' and change form of 'Register'	Hi Aris, This request is from Ray. Please login to London Spicer OAR: https://londonspicer-oar.rschooltoday.com
47200	OAR - New London Spicer - Peg Peterson locked out and cannot access OAR	Hi Aris and Devs, Please unblock Peg's login so she can access OAR. Thanks Regards,BiancaHi Bianca, Unblock t
47206	OAR - Absegami High School - wants to un-cancel a registration previous cancelled.	Good day,Our client, Absegami High School, has a registration that was recently cancelled and wants to uncan

Gambar 4.1 Contoh tampilan data keluhan klien

4.3.1 Pra-proses Teks

Terdapat beberapa tahap proses yang dilakukan dalam pra-prosesteks. Secara umum, tahap pra-proses teks dibagi menjadi beberapa bagian yaitu proses *tokenization, casefolding,filtering* seperti yang sudah dijelaskan pada Bab II.

Tabel 4.2 Cuplikan Keluhan yang Dicatat Customer Services

Our client, Harlan Community School Districts, would like to move the registrations of High School students under "Activity Registrations" web form to Activity Registration - High School web form AND, registrations of Middle School Students to Activity Registration - Middle School web form in their AR
Our client, New London Spicer, has an AR registration where in the Middle Name and Last Name were interchanged.
Delaware Valley Regional High School, has students who has duplicate registration in AR.
Absegami High School, has a registration that was recently cancelled and wants to uncancel it. Is this possible?
Please change the name of this student in OAR:
Please merge Chelsea's name on the 'Registration' dropdown and connect the ID

to AS.

4.3.1.2 Tokenization

Proses yang paling awal dilakukan yaitu *tokenization*. Pada prinsipnya, *tokenization* adalah proses pemisahan teks menjadi potongan kata yang disebut *token. Tokenization* dilakukan untuk mendapatkan token atau potongan kata yang akan menjadi entitas yang memiliki nilai dalam penyusunan matriks dokumen pada proses selanjutnya. Langkah transformasi proses *tokenization* ditunjukkan pada tabel di bawah ini

Tabel 4.3 Tabel Tokenization

Please merge Chelsea's name on the 'Registration' dropdown and connect the ID to AS.	Please merge Chelsea’s name on the ‘Registration’ dropdown and connect the Id to AS

4.3.1.3 Case Folding

Setelah melalui proses *tokenization*, selanjutnya dilakukan *case folding*. *Case folding* merupakan proses pengubahan huruf dalam dokumen menjadi satu bentuk, misalnya huruf kapital menjadi huruf kecil dan sebaliknya. Perubahan yang terjadi dalam proses *case folding* ditunjukkan pada

Tabel 4.4 Tabel Case Folding

Please merge Chelsea's name on the 'Registration' dropdown and connect the ID to AS.	please merge chelsea’s name on the ‘registration’ dropdown and connect the id to as

4.3.1.4 Filtering

Proses berikutnya *Filtering*, yaitu tahap pengambilan kata-kata yang penting dari hasil *case folding*. Bisa menggunakan algoritma *stop list* (membuang kata yang kurang penting) atau *word list* (menyimpan kata yang penting). Pada proses ini kata dan tanda baca yang tidak memiliki arti yang signifikan atau termasuk *noise* (pengganggu) akan dieliminasi. Kata atau frase yang tidak bermakna secara signifikan, misalnya hashtag (#), url, tanda baca tertentu (*emoticon*). Contoh eliminasi adalah sebagai berikut

Tabel 4.5 Tabel Filtering

Based on Activity id, level id, and season id in AR and AS DB, registered activity is "Dance Team Var/JV - Winter 2014-15" (not Dance Team Var - Winter 2014-15). "Dance Team Var/JV - 2014-15" does not exist in AS C/NC and AR Activity List. Now AR will keep this activity stored in the AR Activity List DB to handle blank activity in FA if the activity no longer exists in AS C/NC.	based on activity id, level id, and season id in ar and as db, registered activity is dance teamv var jv winter 2014 15 not dance team var winter 2014 15 dance team jar jv 2014 15 does not exist in as c nc and ar activity list now ar will keep this activity stored in the ar activity list db to handle blank activity in fa if the activity no longer exists in as c nc

Setelah melewati ketiga tahap pada bagian awal pra-proses tersebut, data sudah bisa dikatakan bersih dan siap olah. Berikut ini adalah cuplikan kumpulan kata yang telah melewati bagian awal pra-pemrosesan.

Tabel 4.6 Tabel Hasil Awal Pra-pemrosesan

our client harlan community school districts would like to move the registrations of high school students under activity registrations web form to activity registration high School web form and, registrations of middle school students to activity registration middle school web form in their ar
our client new london spicer has an ar registration where in the middle name and last ame were interchanged
delaware valley regional high school has students who has duplicate registration in ar
absegami high school has a registration that was recently cancelled and wants to uncancel it is this possible
please change the name of this student in ar

4.4 Pemodelan

Proses *training* dan *testing* dilakukan dengan pembobotan *term frequency* dikombinasikan dengan n-gram. Hasil perhitungan akurasi dapat dilihat pada tabel di bawah.

Tabel 4.7 Hasil Perhitungan Model Klasifikasi

Fitur	**Naive Bayes**	**SVM**
Unigram	0.97	0.73
Bigram	0.98	0.59
Trigram	0.6	0.97

Berdasarkan Tabel 4.7, hasil percobaan menunjukkan bahwa model yang dibangun dengan algoritma Naive Bayes menggunakan fitur bigram dan *term frequency* memiliki nilai akurasi yang paling tinggi yaitu 0,98. Sementara itu, model yang dibangun dengan algoritma SVM terbaik 0,97 dengan menggunakan fitur trigram. Pada penggunaan algoritma Naïve Bayes, fitur bigram memberikan hasil akurasi yang paling baik dibanding fitur unigram maupun bigram. Sementara itu, pada model klasifikasi yang dibangun dengan menggunakan SVM, fitur trigram memberikan hasil yang paling bagus diantara unigram maupun bigram

4.5 Evaluasi

Hasil perhitungan evaluasi menggunakan *precision*, *recall*, dan *f-score* ditunjukkan pada tabel di bawah ini.

Tabel 4.8 Perhitungan *precision*, *recall*, dan *f-score*

Fitur	**Naive Bayes**			**SVM**		
	Prec	**Rec**	**F-score**	**Prec**	**Rec**	**F-score**
Unigram	0,989	0,97	0.97	0.78	0.73	0.73
Bigram	0,97	0,97	0.97	0.81	0.62	0.70
Trigram	0,97	0,989	0.98	0.83	0.64	0.71

Berdasarkan hasil yang diperlihatkan pada Tabel 4.8, diketahui bahwa perolehan f-score dengan algoritma Naive Bayes memberikan hasil yang lebih tinggi dibandingkan dengan algoritma SVM. Nilai f-score dengan algoritma Naive Bayes diketahui mencapai 0,97 untuk fitur unigram dan bigram Sedangkan pada fitur trigram, nilai f-score mencapai 0,98. Perolehan f-score dengan algoritma SVM dengan fitur unigram diketahui memiliki nilai tertinggi yaitu 0,75. Adapun nilai f-score menggunakan fitur bigram yaitu 0,70 dan fitur trigram diperoleh sebesar 0,72. Pada pembahasan selanjutnya, pembahasan difokuskan pada hasil perhitungan *precision* dan *recall* untuk masing-masing fitur dan label. Lebih lanjut, pembahasan juga hanya difokuskan pada hasil perhitungan untuk model klasifikasi yang dibangun dengan algoritma Naïve Bayes yang memiliki hasil pengujian yang lebih baik dibandingkan dengan SVM.

4.5.1 Perhitungan *Precision* dan *Recall* untuk Tiap Fitur dan Kelas

Tabel 4.9 sampai dengan Tabel 4.11 masing-masing menyajikan hasil perhitungan *precision* dan *recall* untuk masing-masing kelas label. Secara berturut-turut, Tabel 4.9, 4.10, dan 4.11 menunjukan hasil perhitungan untuk fitur unigram, bigram dan trigram dari model klasifikasi yang dibangun dengan algoritma Naïve Bayes.

Tabel 4.9 Tabel Perhitungan *Precision* dan *Recall* Untuk Fitur Unigram

Kelas	**Precision**	**Recall**
Registration Form	0.967	0,989
Setup	1	0,989
Connection to AS	0,948	1
Family Account	0,989	0.967

Pada Tabel 4.9, didapatkan tiga kelas dengan menggunakan fitur unigram yang memiliki nilai *recall* lebih besar dari nilai *precision*, yaitu *Registration Form*, *Setup* serta *Connection to AS*. Dalam tabel tersebut, juga didapatkan nilai *precision* yang sempurna untuk *Registration Form* dan nilai *recall* yang sempurna untuk kelas *Setup*. Hal pertama menunjukkan bahwa model

klasifikasi yang dibangun tidak membuat kesalahan satupun dalam mengklasifikan keluhan yang masuk kelas *Connection to AS*. Hal kedua menunjukkan bahwa model klasifikasi yang dibangun bisa dengan tepat memberikan label *Setup* kepada data yang sesuai.

Tabel 4.10 Tabel Perhitungan *Precision* dan *Recall* Untuk Fitur Bigram

Kelas	**Precision**	**Recall**
Registration Form	0.978	0.989
Setup	0.959	0.989
Connection to AS	0.989	0.979
Family Account	1.0	0.979

Berdasarkan hasil pada Tabel 4.10, didapatkan dua kelas dengan fitur bigram yang memiliki nilai *recall* lebih besar dari nilai *precision*, yaitu *Registration Form* dan *Setup* dan juga dua kelas dengan nilai *precision* yang lebih besar yaitu *Connection to AS* dan *Family Account*. Dalam tabel tersebut, didapatkan pula nilai *precision* yang sempurna untuk *Family Account*. Hal ini menunjukkan bahwa model klasifikasi yang dibangun dengan menggunakan fitur bigram tidak membuat kesalahan satupun dalam mengklasifikan keluhan yang masuk.

Tabel 4.11 Tabel Perhitungan *Precision* dan *Recall* Untuk Fitur Trigram

Kelas	**Precision**	**Recall**
Registration Form	0,979	0.968
Setup	1.0	0.978
Connection to AS	0.979	0.978
Family Account	1.0	1.0

Tabel 4.11 menunjukkan bahwa dengan menggunakan fitur trigram didapatkan hampir semua kelas memiliki nilai *precision* yang lebih besar dari nilai *recall*. Dalam tabel tersebut diperoleh nilai *precision* yang sempurna untuk *Setup* dan *Family Account*. Penggunaan fitur trigram dalam studi kasus keluhan memberikan model klasifikasi akurasi yang sempurna dalam

mengklasifikasikan keluhan yang termasuk kelas *Family Account*. Dengan nilai *precision* dan *recall* yang sempurna, maka tidak satupun data yang seharusnya termasuk kelas tersebut diklasifikasikan ke kelas yang lain.

4.6 Penerapan Model

Pembahasan ini adalah hasil penerapan model klasifikasi yang dibangun dengan algoritma Naïve Bayes yang memiliki hasil pengujian yang lebih baik dibandingkan dengan SVM. Berikut ini adalah hasil dari penerapan model untuk mengetahui jumlah keluhan terbanyak. Masing-masing kelas terdapat 4 kombinasi kalimat terbanyak

Tabel 4.12 Penerapan Pada Kelas Registration Form

Kelas Registration Form	**Jumlah Kemunculan**
cant + autopopulate	25
blank+field	10
retrieved + data	7
request+to+remove	8

Pada tabel 4.12 kelas Registration Form maka keluhan yang paling muncul terkait dengan *cant+autopopulate*. Hal ini berkaitan dengan gagalnya pengisian nomor studi secara otomoatis.

Tabel 4.13 Penerapan Pada Kelas Setup

Setup	**Jumlah Kemunculan**
please+setup	20
school+site+address	15
payment+credit+account	15
add+feature	10

Pada tabel kelas 4.13 Setup maka keluhan yang paling sering muncul berkaitan dengan *please+setup*. Hal ini tidak berkaitan dengan keluhan akan tetapi terakit dengan pengaitan koneksi antar situs sekolah.

Tabel 4.14 Penerapan Pada Kelas Connection to AS

Family Account	**Jumlah Kemunculan**
disable+connection	2
final+clearance	7
payment+credit+account	16
syncronize+schedule	25

Pada tabel 4.14 kelas Connection to AS maka keluhan yang paling muncul berkaitan dengan *syncronize+schedule*. Hal ini berkaitan dengan koneksi yang mensinkronasi jadwal.

Tabel 4.15 Penerapan Pada Kelas Family Account

Family Account	**Jumlah Kemunculan**
merge +parent+student	30
disable+student+parent	12
payment+credit+account	4
add+credit+card	4

Pada tabel 4.15 kelas Family Account maka keluhan yang paling muncul berkaitan dengan *merge+parent+student*. Hal ini berkaitan dengan penggabungan akun *parent* dan akun *student*.

BAB V
Kesimpulan

5.1 Kesimpulan

Pada penelitian ini telah berhasil dibangun model untuk melakukan klasifikasi keluhan pelanggan produk perangkat lunak. Secara garis besar, terdapat dua jenis model yang dibangun dengan dua pendekatan yang berbeda yaitu Naive Bayes dan Support Vector Machine. Berdasarkan hasil eksperimen, model SVM memiliki akurasi yang lebih rendah dengan perbedaan yang cukup signifikan jika dibandingkan dengan model yang dihasilkan dari algoritma Naive Bayes. Hal tersebut kemungkinan disebabkan algoritma SVM yang digunakan dalam percobaan ini adalah algoritma SVM dengan linear kernel.

Linear kernel pada SVM hanya berjalan optimal untuk kasus klasifikasi biner (binary classification), sedangkan data keluhan yang digunakan dalam percobaan ini memiliki lebih dari satu jenis kelas label sehingga dikelaskan sebagai *multiclass classification*. SVM akan berjalan optimal pada *multiclass classification* dengan menggunakan kernel yang didesain untuk data multidimensi. Hal tersebut diluar cakupan dari percobaan ini dan akan menjadi bagian dalam penelitian selanjutnya.

Pada perhitungan *precision*, *recall*, dan *f-score* diketahui bahwa hasil perhitungan ketiganya memiliki nilai yang sama dengan perhitungan akurasi. Secara keseluruhan, hasil perolehan *f-score* dengan algoritma Naive Bayes memberikan hasil yang lebih tinggi dibandingkan dengan algoritma SVM. Model klasifikasi yang dibangun dengan Naïve Bayes memiliki nilai tertinggi ketika menggunakan fitur trigram, sementara model klasifikasi yang dibangun dengan SVM memiliki nilai tertinggi ketika menggunakan fitur unigram.

Kesimpulan lain dari penelitian ini tentang keluhan yang paling sering muncul antara lain pada kelas Registration Form maka keluhan yang paling muncul terkait dengan *cant+autopopulate*. Pada kelas Setup maka keluhan yang paling sering muncul berkaitan dengan *please+setup*. Hal ini tidak berkaitan dengan keluhan akan tetapi terakit dengan pengaitan koneksi antar situs sekolah. Pada kelas Connection to AS maka keluhan yang paling muncul berkaitan dengan *syncronize+schedule*. Hal ini berkaitan dengan koneksi yang mensinkronasi

jadwal. Pada kelas Family Account maka keluhan yang paling muncul berkaitan dengan *merge+parent+student*. Hal ini berkaitan dengan penggabungan akun *parent* dan akun *student*.

5.2 Saran

Penelitian mengenai *text mining* merupakan salah satu penelitian yang sedang berkembang pesat saat ini seiring dengan berkembangnya teknologi digital yang banyak menghasilkan informasi berupa data tekstual. Data awal yang dipakai pada penelitian ini masih berupa data berformat sql dalam Bugzilla dan bercampur dengan beragam data keluhan dari produk lain selain produk *Activity Registration* sehingga tahap pra-proses cukup sulit dilakukan, membutuhkan waktu yang cukup lama dan lebih banyak untuk memisahkan dokumen yang berguna. Saran peneliti bagi penelitian-penelitian selanjutnya antara lain:

1. Data penelitian lebih baik diambil dari data yang sudah dipisahkan secara jelas dan spesifik atau tidak bercampur dengan data dari produk lain
2. Penelitian di masa depan dapat mempertimbangkan pemilihan fitur lain untuk mengelompokkan kata misalnya dengan memperhatikan sinonim, akronim, dan prinsip kedekatan kata sehingga hasil yang didapatkan akan lebih optimal.
3. Data bisa diambil dari *bug tracker* lain, jadi tidak hanya Bugzilla saja. Namun tentunya hal ini juga akan proses yang berbeda pula.

Daftar Pustaka

Abbot, D 2013. Introduction to Text Mining : Virtual Data Intensive Summer School. *Abbot Analytics, Inc.*

Antoniol, G.D., M. Gall, H. Pinzger, M. 2004. Towards the integration of versioning systems, bug reports and source code meta-models. *Elsevier.*

Beard, R. 2014. 9 Costumer Feedback Software Tools: Comparison & Review. *Diakses tanggal 26 Oktober 2015 dari* http://blog.clientheartbeat.com/customer-feedback-software/

Bidgoli-Minaei, B & Akhondzadeh, E. 2010. A New Approach of Using Association Rule Mining in Customer Complaint Management. *IJCSI,* 141

Botha, G. R. & Barnard, E. 2012. Factors that affect the accuracy of text-based language identification. *Computer Speech and Language Vol. 26* , 307–320.

D'Ambros, M & Lanza, M. 2006. Software bugs and evolution: A visual approach to uncover their relationship. *IEEE*.

Dave, K., Lawrence, S., Pennock, D.M. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. *Proceedings of the 12th international conference on World Wide Web,* 519-528

Dumbill, E. 2014. Forbes Magazine. *Defining Big Data*. Diakses 3 Maret 2015 dari http://www.forbes.com/sites/edddumbill/2014/05/07/ defining-big-data/

Feldman, R. & Sanger, J. 2007. The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. *Cambridge University Press.*

Fischer, M & Gall, H. 2004. Visualizing feature evolution of large-scale software based on problem and modification report data. *Journal of Software Maintenance and Evolution: Research and Practice.*

Furnkranz, J. 2009. A Study Using N-Gram Features for Text Categorization. *Austrian Research Institute for Artificial Intellegence*

Gegick, M., Rotella, P. & Xie, T. 2010. Identifying Security Bug Reports via Text Mining: An Industrial Case Study. *IEEE.*

Gullo, F. 2015. From Patterns in Data to Knowledge Discovery: What Data Mining Can Do. *Physics Procedia 62* , halaman 18-22.

Jiawei, H., Kamber, M., & Pei, J. 2012. Data Mining: Concepts and Techniques Third Edition. *Waltham, MA: Morgan Kaufmann.*

Kumar, V. 2009. Text Mining, Classification, Clustering, and Applications. *CRC Press.*

Lancaster, F.W. 2011. The Measurment and Evaluation of Library Service. *Arlington: Information Resources Service.*

Linoff, G. S., & Berry, M. J. 2011. Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management Third Edition. *Indianapolis, IN: Wiley Publising, Inc.*

Manning, C., D., Raghavan, P., & Schutze, H. *2009*. An Introduction to Information Retrieval. *Cambridge University Press Online Edition.*

McLeod, R. & Schell, G.P. *2007*. Management Information Systems, *edisi ke-10. Pearson Prentice Hall, New Jersey.*

Miner, G., Delen, D., Elder, J., Fast, A., Hill, T., & Nisbet, R. 2012. Practical Text Mining and Statistical Analysis for Non-Structured Text Data Applications. *Oxford: Elsevier.*

Mooney, R. J. 2006. CS 391L Machine Learning Text Categorization. *University of Texas, Austin.*

Ordenes, F.V., Ludwig, S., De Ruyter, K, Grewal & Wetzels, D.2014. Analyzing customer experience feedback using text mining: A linguistics-based approach. *Journal of Service Research 17 (3)*, 278-295

Pfahringer , B. 2006. A semi-supervised Spam mail detector. *Department of Computer Science, University of Waikato, Hamilton, New Zealand.*

Robertson, S. 2004. Understanding Inverse Document Frequency:On theoretical arguments for IDF. *Journal of Documentation 60 no. 5,* halaman 503–520

Santosa, B. 2007. Data Mining: Teknik Pemanfaatan Data untuk Keperluan Bisnis.*Yogyakarta: Graha Ilmu.*

Saputra, I. P. A. 2011. Penggunaan Algoritma TFIDF Dalam Proses Hierarchial Template Maching. *Konferensi Nasional Sistem dan Informatika, Bali*.

Software Testing Help. 2015. Diakses pada tanggal 26 Juni 2015 dari http://www.softwaretestinghelp.com/popular-bug-tracking-software/

Sokolova, M., & Lapalme, G. 2009. *A systematic analysis of performance measures for classification tasks*. Information Processing and Management 45, 427-437.

Suh, J. H., Park, C. H. & Jeon, S. H. 2010. Applying text and data mining techniques to forecasting the trend of petitions *file*d to e-people. *Expert Systems with Applications*,37 ,7255-7268.

Sun, B., Li, S. & Wilcox, R.T. 2011. Cross-selling ordered poducts: An application to consumer banking services . *Journal of Marketing Researach ,* 42,233-239.

The Bugzilla Team. 2015. Bugzilla Documentation Release 5.0rc2+ *diakses pada tanggal 21 Maret 2015.*

Wu , X., & Kaiser, Pa. 2011. BUGMINER:Software Reliability Analysis Via Data Mining of Bug Reports. *Diakses tanggal 30 Januari 2015 www.ksi.edu/seke/Proceedings/seke11/55_Leon_Wu.pdf.*

Wu, X & Kumar,V. 2007. The Top Ten Algorithms in Data Mining. *Chapman and Hall.*

Xiang, Z., Schwartz, Z., Gerdes Jr, J. H., & Uysal, M. 2015. What can big data and text analytics tell us about hotel guestexperience and satisfaction?. *International Journal of Hospitality Management 44* halaman 120-130

Zhai, Z.,Liu, B., Xu, H., Jia, P. 2011. Clustering Product Features for Opinion Mining. *Elsevier*

Zhai, C., & Aggarwal, C. C. 2012. Mining Text Data. *New York: Springer*