BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu menanyangkut topik pembahasan yang di angkat sangat berguna untuk dijadikan sebagai dasar atau refrensi dan hubungan antar penelitian terdahulu dengan penelitian yang saat ini dilakukan, sehingga hal yang bersifat duplikasi dapat dihindari. Selain daripada itu untuk tinjauan pustaka sendiri juga bermanfaat untuk mengetahui arti penting yang bisa memberikan kontribusi pada perkembangan ilmu pengetahuan.

Dalam hal ini penulis bertujuan untuk mendukung penelitian yang sedang dilakasanakan. Selain untuk memaparkan gagasan atau argumen diperlukan data yang relevan dengan penelitian, sehingga dapat dibilang *valid*. Tujuan yang lain dilaksanakannya studi terhadap penelitian terdahulu adalah untuk mencari atau menggali kekurangan dan kelebihan suatu metode yang akan di pergunakan. Dengan ini peneliti dapat menghindari permasalahan terkait.

Oleh karena itu sebelum melaksanakan penelitian maka perlu dilakukan eksplorasi dan studi pustaka baik melalui jejaring ineternet maupun jurnal yang relevan dengan topik yang di angkat, yang dimaksud dalam hal ini adalah masalah penggalian data opini atau Analisis Sentimen. Berikut ini merupakan penelitian-penelitian yang berhubungan dengan data dan metode yang dipergunakan.

Pertama yaitu penelitian ang pernah dilakukan oleh (Rachmat C & Lukito, 2016) tentang penggunaan *Naïve Bayes* guna melakukan klasifikasi sentimen apakah positif atau negatif terhadap komentar dari postingan kampanye politik dari

halaman *Facebook*. Yang dipakai di dalam penelitian ini adalah studi kasus komentar dan status terhadap halaman *Facebook* calon presiden Republik Indonesia pada pemilihan umum tahun 2014. Tahapan yang dilakukan untuk penelitian ini yaitu dengan pengumpulan data 68 status yang berisi 3.400 komentar selama masa kampanye, dengan kegiatan *Preprocessing* tokenisasi, *stemming*, pembobotan token, selanjutnya dilakukan proses klasifikasi , dan menggunakan *confusion matrix*. Dari hasil pengujian dan implementasi yang telah dilakukan, metode *Naïve Bayes* mempunyai tingkat akurasi klasifikasi sentimen hingga mencapai lebih dari 83%.

Penelitian yang dilakukan oleh (Indrawan, Maharani, & Kurniati, 2012) dengan judul "Analisis Sentimen Berdasarkan Fitur Produk Menggunakan *Opinion Lexicon* dan *Wordnet*" menghasilkan keluaran ringkasan dalam bentuk data teks yeng terdiri dari kalimat opini atau pendapat yang sudah di kategorikan berdasarkan fitur produk dan orientasinya. Penelitian ini digunakan dengan 3 langkah, yaitu :

1. Melakukan pengekstrakan dan proses identifikasi fitur produk dari *review* kostumer *(fiture extraction).*
2. Melakukan indentifikasi kalimat opini yang mengandung fitur produk guna ditentukan orientasinya *(sentiment analysis).*
3. Ringkasan yang di hasilkan yaitu berupa klasifikasi opini berdasarkan fitur produk *(summary generation).* Data Uji yang dipergunakan berasal dari *review* kostumer di situs *Amazon.com* karena mempunyai informasi yang lengkap dan detail.

Penelitian tentang analisis sentimen terhadap toko *online* pada sosial media sebelumnya pernah dilakukan oleh (Gusriani, Wardhani, & Zul, 2016). Data yang didapat dari halaman akun toko *online Facebook* Berrybenka. Metodologi ini dimanfaatkan untuk menganalisis sentimen dimulai dari *collecting* data, *prepocessing*, *feature selection*, pengukuran akurasi dan klasifikasi menggunakan metode *Naïve Bayes*. Hasil dari analisis menunjukkan kestabilan akurasi sesudah diuji dengan *k-Fold Cross Validation* serta dengan *Confusion Matrix* dengan tingkat akurasi hingga 93.7% dimana hasil minimum *support* untuk *Frequent Itemset Mining* sampai 0.014.

(Kristiyanti, 2015) melakukan penelitian analisis sentimen *review* produk kosmetik memakai algoritma *support vector machine* dan *particel swarm optimization*. Penelitian yang menggunakan penggabungan metode pemilihan fitur, yakni *particle swarm optimization* supaya dapat meningkatkan pengklasifikasian akurasi *support vector machine*. Penelitian ini menghasilkan teks berbentuk positif atau negatif dari ulasan produk kosmetik. Berdasarkan pengukuran tingkat akurasi dari *support vector machine* sebelum dan sesudah penambahan metode pemilihan fitur. Evaluasi dilakukan meggunakan 1 *fold cross validation*, sedangkan untuk pengukuran akurasinya sendiri diukur menggunakan *Confusion Matrix* dan karva ROC. Hasil penelitian menghasilkan tingkat akurasi *support vector machine* dari 89.00% menjadi 97.00%.

Penelitian tentang klasifikasi opini masyarakat pada *Twitter* pemerintahan kota Surabaya sudah dilakukan oleh (Faradhillah, Kusumawardani, & Hafidz, 2016). Data yang didapat dari akun *Twitter* @e100ss dan @SapawargaSby selama periode

1 September 2015 sampai 13 Oktober 2015, dikategorikan menjadi 3 kelas yaitu sentimen negatif, positif, dan netral. Pengklasifikasian terbaik didapatkan oleh model memakai algoritma *Support Vector Machine* dengan hasil akurasi klasifikasi sebesar 79.81%. Selanjutnya dibuat *web framework* guna memvisualisasikan berupa *wordcloud* dan sebuah grafik *streamgraph* yang divisualisasikan secara interaktif dengan aplikasi *RShiny*.

Analisis sentimen menggunakan metode *Naïve Bayes* dalam penelitian terdahulu pernah dilakukan oleh (Nugroho, Chrisnanto, & Wahana, 2016). Masyarakat mengeluarkan beragam opini di media sosial tentang pelayanan dari transportasi *online* dengan jumlah yang besar, sehinga muncul kesulitan suntuk menentukan opini yang bersifat negatif, positif, atau netral. Sehingga penelitian yang dilakukan ini bertujuan untuk membangun sistem yang mempu membuat proses pengklasifikasian sentimen ke dalam sentimen positif, negatif, dan menalurkan opini tersebut ke setiap jasa yang bersangkutan dengan opini yang muncul. Studi kasus yang digunakan merupakan ojek *online* dan data yang bersumber dari komentar mention akun @*gojekindonesia* dan @*GrabID*. Hasil yang didapatkan dari proses akurasi *Naïve Bayes* mendapatkan ketepatan hingga sampai 80%.

Tabel 2. 1 Penelitian terdahulu

No.	Peneliti	Judul Penelitian	Metode	Tujuan
1.	Antonius Rachmat C, Yuan Lukito (2016)	Klasifikasi Sentimen Komentar Politik dari *Facebook* Page Menggunakan *Naïve Bayes*	*Naïve Bayes*	Menjawab tingkat keakuratan metode *Naïve Bayes* dalam menentukan klasifikasi sentiment terhadap data komentar pengguna media sosial *Facebook* terhadap data di Pemilu Presiden Republik Indonesia 2014
2.	Bani Pramadhana Indrawan, Warih Maharani, Angelina Prima Kurniati (2012)	Analisis Sentimen Berdasarkan Fitur Produk Menggunakan *Opinion Lexicon* Dan *Wordnet*	*Opinion Lexicon* dan *Wordnet*	Mengklasifikasikan opini positif dan negatif berdasarkan fitur produk
3.	Syahmia Gusriani, Kartina Diah Kusuma Wardhani, Muhammad Ihsan Zul (2016)	Analisis Sentimen Terhadap Toko *Online* di Sosial Media Menggunakan Metode Klasifikasi *Naïve Bayes* (Studi Kasus : *Facebook* Page BerryBenka)	Web *Crawling*, *Naïve Bayes*	Mengimplementasikan analisis sentimen guna menentukan kecenderungan pandangan masyarakat publik terhadap pelayanan dan produk kepada pelayanan dan produk suatu toko-toko *online*
4.	Dinar Ajeng Kristiyanti (2015)	Analisis Senimen *Review* Produk Kosmetik Melalui Kamparasi *feature selection*	*Support Vector Machine* dan *Particle Swarm Optimization*	Mengklasifikasikan teks dalam bentuk positif dan negatif dari *review* produk pada kosmetik
5.	Nuke Y. A. Faradhilah, Renny P. Kusumawardani, Irmasari Hafidz (2016)	Eksperimen Sistem Klasifikasi Analisa Sentimen *Twitter* Pada Akun Resmi Pemerintahan Kota Surabaya Berbasis Pembelajaran Mesin	Klasiikasi Teks, *Naïve Bayes*, *Support Vector Machine*	Mengklasifikasikan opini masyarakat publik pada data dari *Twitter* agar Pemerintahan Kota Surabaya meningkatkan kinerjanya
6.	Didik Garbian Nugroho, Yulion Herry Chrisnanto, Agung Wahana (2016)	Analisis Sentimen Pada Jasa Ojek *Online* Menggunakan Metode *Naïve Bayes*	*Naïve Bayes*	Membuatkan sistem yang bisa dan mampu mengklasifikasikan sentimen ke dalam bentuk sentimen positif, negatif, atupun netral

Tabel 2. 2 Penelitian yang dilakukan

No.	Peneliti	Judul Penelitian	Metode	Tujuan
1.	Eko Budi Santoso (2019)	Analisis Sentimen Calon Presiden Indonesia 2019 Berdasarkan Komentar Publik di *Facebook*	*Naïve Bayes*	Mengetahui gambaran umum tentang presepsi pengguna *Facebook* pada calon presiden Indonesia tahun 2019

2.2 Landasan Teori

Pada penelitian yang dilakukan terdapat beberapa teori dasar yang dapat digunakan sebagai acuan dan refrensi terkait pembahasan mengenai ekstraksi data media sosial. Sumber data yang didapat dari sebuah jurnal yang sama dengan penelitian ini serta melakukan studi pustaka dari buku dan internet.

2.2.1 Analisis Sentimen

Hal ini bisa disebut juga dengan *opinion mining*, merupakan suatu bidang ilmu guna menganalisa dari suatu pendapat, sentimen, evaluasi, serta penilaian, sikap dan emosi pada publik terhadap entitas seperti produk atau jasa, organisasi maupun individu, serta masalah, peristiwa, dan atribut (Liu, 2012). Analisis sentimen ini berfokus pada kumpulan opini atau pendapat yang mengungkapkan sentimen kedalam kelas positif atau kelas negatif. Ada beberapa level antara lain :

1. Level dokumen

   Mengklasifikasikan apakah seluruh dokumen opini mengungkapkan sentimen positif atau negatif. Analisis mengasumsikan bahwa setiap dokumen mengungkapkan opini yang objektif tentang suatu entitas tunggal (misalnya, produk tunggal).

2. Level kalimat

   Mengklasifikasikan dari setiap kalimat opini mengungkapkan dan menentukan apakah setiap kalimat menyatakan opini positif, negatif, atau netral.

3. Level entitas dan aspek

   Menemukan sentimen pada suatu entitas dan aspeknya. Sebagai contoh, kalimat "Jalan tol yang sudah dibangun sangat bagus dan indah, tapi untuk pembayaran tol sangat mahal". Ada dua aspek evaluasi, selalu memenuhi janji dan berlangsung sangat lambat, dari seorang pemimpin (entitas). Sentimen pada “Jalan tol yang sudah dibangun sangat bagus dan indah” adalah positif, tapi sentimen pada “tapi untuk pembayaran tol sangat mahal” adalah negatif. Bagus dan indah dan sangat mahal adalah target pendapat.

Analisis sentimen muncul guna mengatasi kondisi apabila terjadi timbul banyaknya akan informasi teks yang tidak terstruktur.

2.2.2 Teks Mining

Text Mining atau penambangan teks adalah bagian dari suatu bidang keilmuan yang hanya khusus pada bagian dari pencarian pola dari informasi yang relevan pada data teks atau dokumen dalam jumlah yang besar. *Text mining* bertujuan untuk memperoleh sebuah informasi yang bisa digunakan dari sekumpulan dokumen. Maka, sumber data yang dipakai dalam *text mining* merupakan kumpulan teks yang mempunyai format tidak terstruktur. Oleh sebab itu, pada proses ini diperlukan pengubahan bentuk dari data yang tidak terstruktur menjadi data yang terstruktur yang mempunyai nilai numerik (Weiss, Indurkhya, & Zhang, 2010).

Setelah data menjadi terstruktur dan mempunyai nilai numerik maka data bisa diproses untuk mengekstrak informasi dari dokumen-dokumen teks yang bisa digunakan untuk kepentingan analisis berbagai bidang multidisipilin yaitu klasifikasi, klasterisasi, informasi temu kembali, visualisasi atau berbagai analis teks yang lainnya (Akilan, 2015).

Saat ini *text mining* telah mendapat perhatian dalam banyak bidang (N. W. Saraswati, 2011), tiga diantaraya sebagai berikut :

1. Aplikasi Media Online

   Text mining saat ini dipakai oleh perusahaan media yang besar, contohnya seperti perusahaan Tribune, untuk mengapus ambigu dalam suatu informasi dan untuk menyampaikan kepada pembaca dengan pengalaman pencarian yang lebih baik. Selain dari itu, pada editor diuntungkan dengan mampu berbagi paket berita, secara signifikan meningkatkan peluang untuk mendapatkan uang dari konten.

2. Aplikasi Pemasaran

   Saat ini pada *Text mining* juga sudah digunakan dalam bidang pemasaran, lebih fokus dalam hal analisis manajemen hubungan dengan pelanggan yang menerapkan suatu model analisis prediksi guna untuk pengurangan pelanggan.

3. *Sentiment Analysis*

   Sentiment Analysis menggunakan analisis dari penilaian pada film guna memperkirakan berapa baik *review* terhadap sebuah film yang di produksi. Analisis ini membutuhhkan sekumpulan data yang berlabel dari efektifitas dari kumpulan kata.

2.2.3 *Preprocessing*

Tahap *Preprocessing* ini digunakan agar bisa mencegah data yang kurang relevan atau sempurna , data yang tidak konsisten. Tahap ini guna membersihkan data dari *noise*, menyamakan bentuk kata dan mengurangi *volume* kata. Supaya pada tahap klasifikasi lebih baik dan optimal dalam proses perhitungannya. Ada beberapa tahap dalam proses *preprocessing* yakni *correcting slang word*, *filtrasi*, *stemming*, dan *tokenisasi*.

1. *Correcting Slang Word*

   Tahap ini adalah ketika penggunaan kata-kata tidak formal dan suatu ekspresi yang kurang baku atau tidak *valid* dalam bahasa tertentu tapi masih diterima dan dipahami dalam pergaulan. *Slang word* perlu dilakukan perbaikan untuk mendapat data informasi yang dapat diproses. Contohnya adalah "yg", "smga", "krna", "bsa" dan lain-lain.

2. *Filtrasi*

   Tahap ini adalah suatu tahapan pengambilan kata-kata yang perlu dan penting dari hasi token. Dan tahhap ini memerlukan Algoritma *stoplist* (membuang kata yang tidak perlu dan kurang berpengaruh) atau *wordlist* (kumpulan kata-kata penting ang tersimpan). Sedangkan *stopword* adalah suatu kumpulan kata yang kurang perlu dan bersifat deskriptif dari suatu dokumen sehingga mampu dan dapat dilakukan pembuangan kata. Contohnya adalah "saya", "dan", "kamu", dan seterusnya (Putri, 2016). Dalam tahap filtrasi ini memakai *stopword/stoplist* agar kumpulan kata yang sering muncul dan buka merupakan kata yang penting dalam sebuah dokumen dapat dihilangkan tanpa

mempengaruhi dari makna dan arti dalam dokumen yang akan diproses ke tahapan selanjutnya.

3. *Stemming*

Tahap ini berguna dalam proses pengolahan kata guna mendapatkan kata dasar dari suatu kata yang memiliki imbuan dengan asumsi bahwa kumpulan kata tersebut sebenarnya mempunai makna dan arti yang sama. Algoritma ini bekerja berdasarkan struktural morfologi didalam kalimat bahasa Indonesia, yang terdiri dari awalan, akhiran, dan sisipan. Tujuan dari tahap ini adalah :

1) Dalam perkara keefisiensian, pada *stemming* dilakukan pengurangan jumlah kata dalam dokumen agar mengurangi kebutuhan dalam ruang penyimpanan dan mempercepat dalam melakukan pencarian.
2) Dalam perkara keefektifan, *stemming* dilakukan untuk mengurangi *recall* dengan pengurangan bentuk-bentuk kata ke dalam bentuk dasarnya (Putri, 2016). Sebagai contoh adalah kata "duduk-lah", "minum-lah", "jika-pun" dan sebagainya.

4. Tokenisasi

Pada tahap ini dalam proses pemecahan kalimat menjadi kumpulan kata yang lebih bermakna. Tahapan pertama kali yang dilakukan yakni dengan proses normalisasi kata dengan merubah setiap karakter huruf menjadi huruf kecil. Proses awal dengan menghilangkan *delimiter* seperti tanda baca dan simbol yang ada dalam suatu dokumen tersebut seperti tanda (,),*,$,@,!,?,/ dan lain sebagainya. Selanjutnya proses penguraian teks yang awalnya berupa kumpulan dari sebuah kalimat yang berisi kata-kata. Proses pemotongan *string*

berdasarkan tiap kata yang menyusunnya, seperti halnya spasi, proses tokenisasi ini mengandalkan karakter spasi pada dokumen teks guna melakukan pemisahan. Hasil dari proses ini hanyalah kumpulan kata saja (Putri, 2016).

2.2.4 Fitur dan Pembobotan

Proses Pembobotan adalah sebuah metode guna merubah inputan data menjadi suatu fitur vektor. Metode yang sering dan umum dipakai adalah *bag-of-*fitur*e*. Contohnya terdapat sederet fitur seperti pada vektor $\{f_1, f_2, \dots f_n\}$ dimana sekumpulan fitur sebanyak n yang telah ditentukan sebelumnya. Misal kata "puas" maka fitur vektor dari data adalah vektor.

1. *Term Presence*

   Metode ini adalah metode pembobotan pada sebuah dokumen berisi teks yang melihat keberadaan dari daftar kata-kata (*term*) atau fitur yang teradapat pada corpus terhadap suatu dokumen. Apabila suatu fitur yang terdapat pada daftar fitur acuan yang ada pada dokumen yag sedang diboboti maka nilai fitur tersebut pada *feature vector* akan diberikan nilai 1 dan tidak menghiraukan jumlah kemunculan fitur tersebut. Apabila fitur tersebut tidak ada dalam dokumen maka diberikan nilai 0 pada *feature space* (O'Keefe & Koprinska, 2011). Rumus yang digunakan untuk menghitung *Term Presence* (TP) dari fitur t_i, pada d_j ditulis dengan notasi (2.1).

$$tp\left(t_i d_j\right) = \begin{cases} 1 & jika\ terdapat\ t_i\ pada\ d_j \\ 0 & jika\ tidak\ terdapat\ t_i\ pada\ d_j \end{cases} \qquad (\mathbf{2.1})$$

2. *Term Frequency*

Term Frequency atau TF adalah metode yang memiliki kesamaan dengan metode *Term Presence* atau TP yang sudah dijelaskan sebelumnya, tetapi yang membedakan diantaranya adalah TF menggunakan menghitung jumlah kemunculan fitur acuan yang terdapat pada sebuah dokumen bukan hanya dari keberadaan fitur tersebut (O'Keefe & Koprinska, 2011). Rumus TF dapat ditulis dalam persamaan (2.2) dengan $\#(t_i, d_j)$ memiliki arti jumlah dari kemunculan fijtur t_i pada dokumen d_j. Contohnya pada suatu fitur berupa kata "bagus" muncul sebanyak 10 kali maka nilai fitur tersebut pada *feature vector* adalah 10.

$$tf\ (t_i d_j) = \#(t_i, d_j) \qquad (\mathbf{2.2})$$

3. *Term Frequency – Inverse Document Frequency* (TF-IDF)

Metode ini biasa disebut juga TF-IDF yang merupakan sebuah algoritma pembobotan yang tersusun dari dua nilai yang berasal dari dua buah algoritma dengan pembobotan yang beda, yakni *Term Frequency* atau TF dan *Inverse Document Frequency* atau IDF. Rumus yang ditulis di persamaan (2.3) memperlihatkan formula perhitungan IDF pada suatu kumpulan dokumen D dengan |D| merupakan jumlah dokumen dan $\#d(t_i)$ merupakan banyakna dari dokumen dimana dari suatu kata (t_i).

$$idf(t_i, d_j) = log \frac{|D|}{\#d(t_i)} \quad (2.3)$$

$$tfidf(t_i, d_j) = tf(t_i, d_j)\ x\ idf(t_i, d_j) \quad (2.4)$$

Keluaran dengan fitur *term* berikut dengan kata yang sering muncul pada dokumen yang akan menghasilkan nilai dari TF-IDF yang sangat tinggi. Sedangkan untuk fitur yang sering muncul pada sebuah dokumen akan menghasilkan nilai yang sangat rendah. Dengan metode ini *term*s atau fitur-fitur yang penting akan mempunyai nilai yang sangat rendah (O'Keefe & Koprinska, 2011).

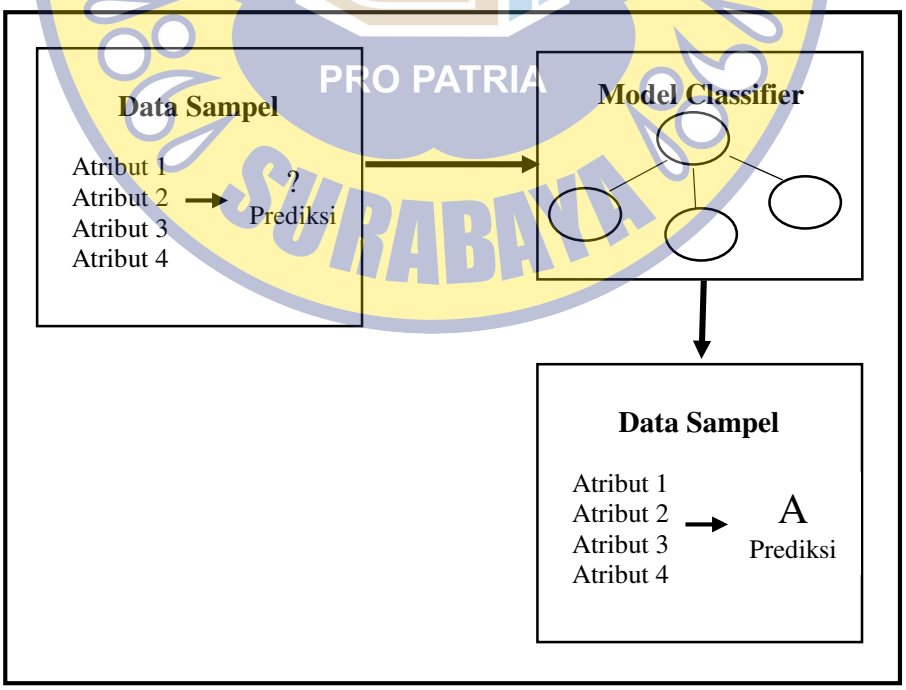
2.2.5 Klasifikasi

Klasifikasi adalah suatu kegitan atau tahapan guna melakukan penilaian terhadap suatu objek data untuk masuk dalam suatu kelas tertentu dari sejumlah kelas yanng tersedia (Prasetyo, 2012). Teknik klasifikasi ini sangat cocok digunakan untuk memprediksi atau menggambarkan kumpulan dari sebuah data dengan kategori biner atau nominal dan namun kurang efektif untuk kategori ordinal. Misal, untuk mengklasifikasikan seseorang dengan kriteria tinggi, menengah dan rendah untuk tingkat penghasilan (Tan, Pang-Ning and Steinbach, Michael and Kumar, 2006).

Untuk data klasifikasi sendiri mempunyai dua tahapan proses. Yaitu tahapan pertama dengan membuat sebuah model yang berdasarkan dari kumpulan data *class* yang disebut dengan kata lain *learned model*. Model dalam hal ini dibangun dengan

cara menganalisa *record database*, dan setiap *record* diibaratkan menjadi *predefined class* yang ditentukan dari suatu atribut yang disebut *class* label atribut. Dalam hal ini berakibat terdapat adanya *class* label maka tahap ini juga dikenal dengan *supervised learning*. Berbeda dengan halnya *unsupervised learning* atau dikenal dengan *clustering* yang tidak memerlukan *class* label. Tahap pertama ini juga disebut sebagai tahap pembelajaran.

Suatu algoritma klasifikasi yang akan membuat sebuah model klasifikasi dengan cara menganalisis data *training*. Tahap pembelajaran bisa juga dilihat sebagai tahap pembentukan fungsi atau pemetaan $\boldsymbol{y = f(x)}$ dimana *y* merupakan kelas hasil dari prediksi dan *x* adalah *record* yang ingin diprediksi *class*-nya. Bagan proses klasifikasi data sampel menggunakan model *classifier* untuk memperoleh hasil prediksi (Han & Kamber, 2006) dapat dilihat pada gambar 2.1.



Gambar 2. 1 Alur proses klasifikasi
(Han & Kamber, 2006)

Beberapa persiapan yang harus dilaksanakan guna memperoleh hasil dari proses klasifikasi yang bagus dan sesuai yang diharapkan diantaranya adalah (Han & Kamber, 2006).

1. Pembersihan Data

   Kegiatan ini dilakukan guna mengurangi data yang kurang tepat didalam data pelatihan, beberapa metode yang dipakai diantaranya adalah dengan teknik *smoothing* untuk menghapus *noise* data, dan melengkapi data yang hilang dan sebagainya.

2. Analisis Relevansi

   Dari beberapa atribut yang akan dipakai guna proses klasifikasi mungkin akan terdapat atribut yang sangat berhubungan kuat antara satu dengan lainnya, kedua atribut ini mempunyai kesaman sehingga menyebabkan proses klasifikasi menjadi tidak optimal, maka salah satu dari atribut ini dapat dihilangkan.

Hasil klasifikasi dan prediksi dapat dievaluasi dengan menggunakan beberapa kriteria (Han & Kamber, 2006).

1) Akurasi

   Hal ini digunakan guna mengetahui kemampuan dari model klasifikasi untuk bisa memberikan ketepatan hasil prediksi.

2) Kecepatan

   Untuk mengetahui kecepatan iterasi guna mendapatkan model klasifikasi dan iterasi memperoleh hasil prediksi.

2.2.6 *Teorema Bayes*

Teorema bayes merupakan *teorema* yang mengacu pada konsep probabilitas bersyarat (Tan, Pang-Ning and Steinbach, Michael and Kumar, 2006). Metode ini adalah metode pendekatan statistik guna melakukan *inferensi* induksi pada pemasalahan klasifikasi. Semisal A dan B merupakan kejadian dalam ruang sampel. (T. Larose, 2006) menyatakan probabiltas bersyarat dalam persamaan (2.5).

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

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

$$P(A|B) = \frac{P(A|B)P(A)}{P(B)} \qquad (\mathbf{2.6})$$

2.2.7 *Naïve Bayes Classifier*

Naïve Bayes Classifier adalah suatu algoritma yang sederhana namun mempunyai kemampuan dan akurasi yang sangat tinggi dan termasuk dalam metode *machine learning* (Rish, 2001). Berdasarkan kompleksitasnya *Naive Bayes* merupakan salah satu algoritma paling sederhana dalam menerapkan aturan *Bayes* disertai beberapa keunggulana yaitu sangat efisien, nenbutuhkan data uji yang sedikit, mudah untuk diimplementasikan dan memiliki akurasi relatif tinggi. Selain

itu, *Naive Bayes* juga memiliki kekurangan diantaranya akurasi akan berkurang jika atribut yang menjadi parameter dalam klasifikasi tidak independen atau *nonparametric-continous* (Rianto, 2016).

Metode klasifikasi *Naive Bayes* menggunakan konsep peluang dalam menentukan kelas dalam dokumen. Metode ini memakai asumsi bahwa dalam suatu dokumen kemunculan kata tidak berpengaruh terhadap kemunculan kata yang lain dan ketidak munculan kata tidak mempengaruhi ketidakmunculan kata yang lain (Rianto, 2016). *Naïve Bayes* adalah teknik prediksi yang berbasis probabilistik sederhana berdasar pada penerapan *teorema Bayes* dengan asumsi independensi yang kuat (N. W. S. Saraswati, 2013). *Naïve Bayes classifier* mengasumsikan ada atau tidaknya suatu fitur tertentu pada sebuah kelas tidak mempengaruhi keberadaan fitur lainnya.

1) Model Probabilitas untuk *Naïve Bayes*

Secara mendasar, model probabilitas untuk sebuah *classifier* adalah model peluang bersyarat.

$$P(C|F_1, \dots, F_n) \qquad (\mathbf{2.7})$$

Dengan menggunakan *teorema Bayes :*

$$P(C|F_1, \dots, F_n) = \frac{P(C)P(F_1, F_2, \dots, F_n|C)}{P(C|F_1, F_2, \dots, F_n)} \qquad (\mathbf{2.8})$$

Dimana, variabel C menjelaskan tentang kelas, sedangkan variabel *F1 ... Fn* menjelaskan tentang karakteristik petunjuk yang diperlukan guna melakukan

klasifikasi. Sehingga rumus tersebut menerangkan bahwa peluang masuknya sampel karakteristik tertentu pada kelas C (*Posterior*) adalah peluang munculnya kelas C (sebelum masuknya sampel tersebut, biasa juga disebut *prior*), dikali dengan peluang kemunculan karakteristik-karakteristik sampel pada kelas C (disebut juga *likelihood*), dibagi dengan peluang kemunculan karakteristik-karakteristik sampel secara global (disebut juga *evidence*). Oleh karena itu, rumus di atas dapat juga ditulis sebagai berikut :

$$Posterior = \frac{PriorxLikelihood}{Evidence} \quad (\mathbf{2.9})$$

Nilai dari *Evidence* selalu tetap teruntuk setiap kelas pada satu sampel. Nilai dari *posterior* sendiri nantinya akan dibandingkan dengan nilai-nilai *posterior* kelas yang lainnya untuk menentukan ke kelas apa suatu sampel akan diklasifikasikan.

2.2.8 *Confusion Matrix*

Penelitian ini menggunakan metode *confusion matrix* dalam proses evakuasi untuk mengetahui hasil akurasi klasifikasi. *Confusion matrix* merupakan salah satu *tool* penting dalam metode evaluasi yang digunakan pada *machine learning* yang biasanya memuat dua kategori atau lebih (D. Manning, Raghavan, & Schütze, 2009).

Setiap unsur matriks menunjukkan jumlah contoh data uji untuk kelas sebenarnya yang digambarkan dalam bentuk baris sedangkan kolom

menggambarkan kelas yang diprediksi. Tabel 2.3 menampilkan sebuah *confusion matrix* untuk pengklasifikasian kedalam dua kelas (Sokolova & Lapalme, 2009).

Tabel 2. 3 *Confusion Matrix*

		True Class	
		Positive	**Negative**
Predicted Class	**Positive**	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Matriks tersebut memiliki empat nilai yang dijadikan acuan dalam perhitungan, yaitu :

a) *True Positive* (TP), ketika kelas yang diprediksi positif dan faktanya positif.

b) *True Negative* (TN), ketika kelas yang diprediksi negatif, faktanya negatif.

c) *False Positive* (FP), ketika kelas yang diprediksi positif dan faktanya negatif.

d) *False Negative* (FN), ketika kelas yang diprediksi negatif, faktanya positif.

Berdasarkan nilai True Negative (TN), False Positive (FP), False Negative (FN), dan True Positive (TP) dapat diperoleh nilai akurasi. Nilai akurasi menggambarkan seberapa akurat sistem dapat mengklasifikasikan data secara benar. Dengan kata lain, nilai akurasi merupakan perbandingan antara data yang terklasifikasi benar dengan keseluruhan data. Nilai akurasi dapat diperoleh dengan Persamaan (2.10), untuk mecari nilai *precision* dengan persamaan (2.11), dan dengan persamaan (2.12) untuk mencari nilai *recall* (Prasetyo, 2012).

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \qquad (\mathbf{2.10})$$

$$Precision = \frac{TP}{TP + FP} \qquad (\mathbf{2.11})$$

$$Recall = \frac{TP}{TP + FN} \qquad (2.12)$$

2.2.9 Asosiasi Teks

Dalam (Ulwan, 2016) disebutkan bahwa istilah korelasi sering sekali digunakan untuk menyatakan suatu hubungan antara dua variabel atau lebih yang bersifat kuantitatif, sedangkan istilah asosiasi sering diartikan keeratan hubungan antara dua variabel atau lebih yang bersifat kualitatif. Tujuan analisis korelasi merupakan guna untuk mencari hubungan variabel bebas (*X*) dengan variabel terikat (*Y*), dengan ketentuan data mempunyai syarat-syarat tertentu (Bustami, Abdullah, & Fadlisyah, 2014). Dapat dilihat pada persamaan (2.13) merupakan persamaan untuk menentukan nilai korelasi.

Pada penelitian ini digunakan pendekatan asosiasi untuk menemukan hubungan antar komentar dari pengguna sehingga mendapatkan informasi yang dapat dijadikan bahan rujukan dalam proses evaluasi untuk memperbaiki elektabilitas calon presiden Indonesia tahun 2019.

$$r = \frac{\mathrm{n}.(\Sigma \mathrm{XY}) - (\Sigma \mathrm{X}).(\Sigma \mathrm{Y})}{\sqrt{\mathrm{n}.\Sigma X^2 - (\Sigma X^2)(n.\Sigma Y^2 - (\Sigma \mathrm{Y})^2)}} \qquad (2.13)$$

Dimana :

n = Banyaknya sampel

Σx = Total Jumlah dari Variabel X

Σy = Total Jumlah dari Variabel Y

Σx^2 = Kuadrat dari Total Jumlah Variabel X

Σy^2 = Kuadrat dari Total Jumlah Variabel Y

Σxy = Hasil Perkalian dari Total Jumlah Variabel X dan Variabel Y