BAB II

TINJAUAN PUSTAKA

2.1 Produksi Kopi

Hasil perkebunan kopi saat ini diusahakan rakyat sebesar 94% dan sisanya diusahakan oleh pemerintah/swasta. Kopi merupakan hasil komoditi perkebunan yang mempunyai nilai ekonomi tinggi di antara produk pertanian lainnya. Produk kopi juga memiliki peran penting sebagai devisa Negara, melainkan juga sumber penghasilan bagi tidak kurang dari satu setengah jiwa petani kopi di Indonesia. Kopi di Indonesia memiliki banyak jenis yaitu kopi arabika, kopi liberika, kopi robusta, dan kopi hibrida. Pada umumnya, perkebunan kopi rakyat ini belum dikelola dengan baik, sehingga muncul beberapa masalah yakni masalah produktivitas terhadap kopi tersebut. Produktivitas yang cukup tinggi akan mencapai produksi kopi yang dialokasikan optimal (Kustiari, 2016).

Produksi kopi Indonesia tidak respon terhadap perubahan harga kopi dan komoditas subtitusi dipasar domestik, tingkat upah dan luas area. Penawaran kopi di Indonesia dipengaruhi oleh tingkat teknologi dan jumlah penawaran setahun sedangkan pengaruh harga kopi dan teh secara statistik tidak berpengaruh nyata. Tanda koefisien peubah teh yang negatif menunjukkan bahwa kopi dan teh di Indonesia merupakan *competiting produc.*Untuk jenis kopi yang sering menjadi perbincangan di sosial media terutama pada *twitter* meliputi kopi pahit, kopi sakit, kopi luwak dan kopi hitam. Dari topik kopi memiliki arti masing-masing. Kopi pahit adalah produk

kopi yang rasanya pahit atau tanpa gula. Kemudian, kopi sakit merupakan hastag yang menjadi tranding topik di media sosial khususnya *twitter*, kopi sakit artinya seseorang akan sakit setelah mengkonsumsi kopi. Untuk kopi luwak adalah produk kopi yang berasal dari biji luwak yang sampai saat ini banyak digemari masyarakat di semua kalangan. Dan kopi hitam adalah produk kopi yang berasal dari biji kopi asli yang berwarna hitam yang biasanya sering diseduh oleh orang dewasa (Pradipta *et al.*, 2017)

2.2 Konsumen Kopi

Menurut Zulfi, dkk (2018), perilaku konsumen yaitu sebagian dari pemahaman terhadap tindakan secara langsung oleh konsumen dalan memperoleh, mengkonsumsi, dan juga menjadikan produk dan jasa yang diperoleh habis. Hasil dari pengolahan bahan minuman dari sector pertanian sangat diminati masyarakat untuk membantu proses metabolisme tubuh, menghilangkan dahaga serta sebagai bentuk kebiasaan masyarakarat tersebut. Salah satu jenis minuman yang popular di kalangan masyarakat saat ini adalah kopi (Zulfi *et al.*, 2018)

Saat ini kopi yang sering disukai dikalangan masyarakat setelah teh. Kopi juga dapat diminum baik dingin maupun panas, jadi sesuai dengan keinginan meminum kopi yang secara turun temurun hingga sekarang. Tingkat konsumsi kopi dalam negeri menurut hasil survei LPEM UI tahun 1989 adalah sebanyak 500 gram/kapita/tahun. Bahkan, pada tahun 2011 tingkat mengkonsumsi kopi di Indoneisa mencapai 800 gram/kapita/tahun. Hal ini dilihat dalam kurun waktu 20 tahun meningkat dalam mengkonsumsi kopi

sebesar 300 gram/kapita/tahun. Kebiasaan masyarakat dalam mengkonsumsi kopi setiap harinya. Serta, adanya peningkatan pada taraf hidup dan pergeseran gaya hidup msyarakat di Indonesia juga mendorong terjadinya pergeseran dalam pola konsumsi kopi pada khususnyaa. Kopi yang disukai di kalangan generasi muda yaitu lebih menyukai minum kopi instan, kopi three in one maupun kopi expresso yang disajikan di café-café. Sedangkan pada masyarakat desa/penduduk golongan tua masih menjadikan kopi tubruk (kopi bubuk) sebagai konsumsi utama (AEKI, 2018)

Mengkonsumsi kopi tidak hanya untuk menghilangkan dahaga, tetapi juga untuk menemani aktivitas kehidupan masyarakat. Misalnya dalam kebiasaan sehari-hari, serta dalam acara kantor. Hal ini menjadi tren untuk mengkonsumi kopi yang terus meningkat. (Suisa *et al.*, 2016). Tetapi sebagian masyarakat suka terhadap kopi karena beberapa alasan. Ada beberapa faktor yang menjadikan masyarakat enggan untuk minum kopi. Faktor tersebut yaitu rasa pahit terhadap kopi, penyebab *maag,* perut kembung dan jantung yang berdebar-debar, bahkan menjadikan susah tidur dan sebagainya. Hal ini tentu membuat prsepsi masyarakat terhadap kopi berbeda-beda (Pradipta *et al.*, 2017)

2.3 Text Mining

Menurut Feldman dan Sanger (2007), pengertian *text mining* adalah proses pengetahuan intensif yang memungkinkan penggunanya berinteraksi dan bekerja dari waktu ke waktu pada sekumpulan dokumen menggunakan berbagai macam analisis. *Text mining* atau *text analytics* adalah istilah dari

cara mengektraksi atau mendeskripsikan sebuah teknologi yang mampu dalam menganalisis sebuah teks semi-terstruktur maupun tidak terstruktur, hal inilah yang membedakannya dengan *data mining* dimana data mining mengolah data yang sifatnya terstruktur. *Text mining* juga sebagai proses untuk menemukan suatu informasi atau tren terbaru yang sebelumnya tidak diketahui dengan memproses dan menganalisis data yang sejumlahnya besar atau *big* data.

Dalam *text mining* pada suatu pola yang diekstrak dari data tekstual yang tidak terstruktur bukan berasal dari suatu *database*. Dalam data *mining* yang diolah yaitu data yang terstruktur dari *warehousing* sehingga lebih mudah diproses oleh mesin/komputer. Pada analisis teks lebih sulit karena teks biasanya hanya digunakan sebagai konsumsi manusia. Ditambah struktur teks yang kompleks, struktur yang tidak lengkap, bahasa yang berbeda, dan arti yang tidak standar. Oleh sebab itu pada umumnya digunakan *Natural Languange Processing* untuk analisis teks yang tidak berstruktur tersebut. Tahapan-tahapan dalam *text mining* secara umum adalah *text preprocessing* dan *feature selection* (Feldman & Sanger, 2007) Dimana penjelasan dari tahap-tahap tersebut adalah sebagai berikut:

2.3.1 Text Preprocessing

Tahap awal yaitu *text processing*, tahap ini melakukan Analisa dari segi sintatik, pada tahap ini merupakan tahap dari proses awal yang berfungsi untuk mengubah data teks yang tidak tersetruktur menjadi data terstruktur. Berikut adalah tahap proses secara umum yang dilakukan dalam tahapan awal untuk

text preprocessing adalah sebagai berikut

a. *Spelling Normalization*

Pada tahap ini yaitu proses perbaikan atau subtitusi kata-kata yang salah eja atau singkatan dalam bentuk tertentu. Subtitusi sebuah kata yang dilakukan untuk menghindari jumlah perhitungan dimensi kata yang melebar. Untuk perhitungan dimensi yang melebar jika ada kata salah eja atau disingkat yang tidak diubah polanya, karena kata tersebut faktanya mempunyai arti dan maksud yang sama akan tetapi akan dianggap sebagai entitas yang berbeda saat proses penyusunan sebuah matriks.

b. *Case Folding*

Case Folding merupakan proses dari penyamaan *case* dalam sebuah dokumen. Dimana pada tahap ini dilakukan untuk memudahkan pencarian. Karena tidak semua dokumen memiliki teks yang konsisten dalam penggunaan huruf kapital. Oleh karen itu, fungsi dari tahap *case folding* dibutuhkan untuk mengkonversi seluruh teks dalam sebuah dokumen menjadi suatu bentuk standar (dalam hal huruf kecil atau *lowercase*)

c. *Tokenizing*

Tokenizing merupakan tahap ketiga dari *preprocessing*. Dimana tahapan ini yakni memecahkan kalimat menjadi sebuah kata-kata untuk menjadikan sebuah kalimat yang lebih bermakna. Tahap awal yang dilakukan adalah normalisasi kata dengan mengubah semua karakter huruf menjadi huruf kecil atau *ti LowerCase.* Pada proses *tokenizing* diawali dengan menghilangkan delimiter-delimiter yakni simbol-simbol, tanda baca

pada teks seperti @, $, &, tanda titik (.), koma (,), tanda tanya (?), tanda seru (!) dan simbol lainnya selain huruf latin antara "a" sampai "z". Lalu tahap berikutnya, yaitu proses penguraian sebuah teks yang awalnya berupa kalimat yang berisi kata-kata. Kemudian dilakukan proses pemotongan string berdasrakan tiap kata yang menyusunnya, umumnya setiap kata akan terpisah dnegan karakter spasi, maka proses *tokenizing* mengandalkan karakter spasi pada dokumen teks tersebut untuk pemisahan kata. Hasil dari proses *tokenizing* adalah beberapa kata atau kumpulan kata saja.

d. *Filtering Stopword*

Filtering stopword adalah tahap dalam mengambil kata-kata penting dari hasil *tokenizing*. Pada tahap ini menggunakan algoritma *stoplist* (membuang kata yang kurang penting) atau *wordlist* (menyimpan kata yang penting) untuk digunakan pada tahap ini. *Stopword* adalah kata-kata yang tidak desktiptif atau bukan merupakan kata penting dari suatu dokumen sehingga perlu dibuang. Contoh dari *stopword* adalah "yang", "dan", "di", "dari" dan seterusnya. Pada tahapan ini menggunakan kamus *stoplist/stopword* agar kata-kata yang kurang penting dan sering muncul pada dokumen harus dibuang dan menyisakan kata-kata yang penting serta memiliki arti agar mudah digunakan ke tahap selanjutnya.

2.3.2 Feature Selection

Pada tahap ini adalah tahap lanjutan dari pengurangan dimensi. Meskipun tahap sebelumnya adalah melakukan penghapusan kata-kata yang tidak penting atau tidak deskriptif (*Stopword*), tidak semua kata didalam

dokumen memiliki arti penting. Sehingga, perlu adanya tahap untuk mengurangi dimensi, pemilihan kata hanya dilakukan pada semua kata yang relevan dan benar untuk dipresentasikan dari isi suatu dokumen tersebut, kata yang dinilai penting dapat dilihat dari intensitas kemunculan dan yang paling informatif dari keseluruhan kata dalam dokumen.

Berikut adalah contoh *text mining* yang dilakukan pada *tweet*. Data *tweet* yang dikumpulkan tersebut, dipilah menjadi dua bagian yaitu tweet yang bersifat sentimen dan tidak bersifat sentimen. Dua jenis *tweet* tersebut dapat dilihat berikut :

Tabel 2.1 Contoh data *Tweet* Sentimen dan Non sentimen

Jenis	*Tweet*
Sentimen	"Kemarin pas turun dari Sumbing, sempet beli bubuk kopi arabika asli Temanggung. Pas diseduh rasanya enak banget \ud83\ude0b\u2615\ufer0f
Non Sentimen	"ingin kopi luwak yang enak ? klik saja https://t.co/tOE7ZnzMVq #Coffe"

Sumber *:* (Nuritha, 2017)

Dari data diatas merupakan data *twitter* yang disimpan dalam bentuk sentimen. Data *tweet* sentiment tersebut lalu dikelompokkan menjadi dua kategori yaitu sentimen positif dan sentimen negatif. Data sentimen ini adalah data mentah yang diperoleh dari tahap *perpocessing* sebelum dilakukan proses klasifikasi sentimen. Berikut contoh sentimen positif dan negatif :

Tabel 2.2 Contoh Data *Tweet* Sentimen Positif dan Negatif

Kategori	*Tweet*
Positif	"@dr_tompi Pisang goreng mantap kayaknya nih sambal serudup kopi Gayo khas Aceh diracik dgn Vietnam Drip ulalala makyuss
Negatif	"Selalu setelah minum kopi perutku sakit, apakah ini tanda lambungku gakuat ? \ud83e\udd14 tapi aku cinta kopi bgt\ud83d\ude41\ude83d\ude41\ude41"

Sumber : (Nuritha, 2017)

Selanjutnya mengumpulkan data *tweet* melalui proses tahapan praproses yang meliputi *case folding tokenizing, stopword removing* dan *stemming.* Ada beberapa contoh dari *preprocessing* dengan hasil luarannya. Ditampilkan pada tabel sebagai berikut :

Tabel 2.3 Contoh Luaran Tahapan *Case Folding*

Tweet	*Luaran*
"Selaluuu setelah minum kopi perutku sakit, apakah ini tanda lambungku gakuat ? \ud83e\udd14 tapi aku cinta kopi bgt\ud83d\ude41\ude83d\ude41\ude41"	selauuu setelah minum kopi perutku sakit apakah ini tanda lambungku gakuat \ud83e\udd14 tapi aku cinta kopi bgt\ud83d\ude41\ude83d\ude41\ude41"

Sumber : (Nuritha, 2017)

Tabel 2.4 Contoh Luaran Tahapan *Tokenizing*

Tweet	*Luaran*
“selaluuu setelah minum kopi perutku sakit, apakah ini tanda lambungku gakuat ? \ud83e\udd14 aku cinta kopi bgt\ud83d\ude41\ude83d\ude41\ude41”	Selauu ; setelah ; minum ; kopi ; perutku ; sakit ; apakah ; ini ; tanda ; lambungku ; gakuat ; \ud83e\udd14 ; aku ; cinta ; kopi ; bgt ; \ud83d\ude41\ude83d\ude41\ude41”

Sumber : (Nuritha, 2017)

Tabel 2.5 Contoh Luaran Tahapan *Filtering Stopword*

Tweet	*Luaran*
Selauu ; setelah ; minum ; kopi ; perutku ; sakit ; apakah ; ini ; tanda ; lambungku ; gakuat ; \ud83e\udd14 ; tapi ; aku ; cinta ; kopi ; bgt ; \ud83d\ude41\ude83d\ude41\ude41”	Selauu ; setelah ; minum ; kopi ; perutku ; sakit ; apakah ; ini ; tanda ; lambungku ; gakuat ; tapi ; aku ; cinta ; kopi ; bgt

Sumber : (Nuritha, 2017)

Tabel 2.6 Contoh Luaran Tahapan *Stemming*

Tweet	*Luaran*
Selauu ; setelah ; minum ; kopi ; perutku ; sakit ; apakah ; ini ; tanda ; lambungku ; gakuat ; tapi ; aku ; cinta ; kopi ; bgt	Selauu ; setelah ; minum ; kopi ; perutku ; sakit ; apakah ; ini ; tanda ; lambungku ; gakuat ; tapi ; aku ; cinta ; kopi ; bgt

Sumber : (Nuritha, 2017)

2.4 Pembobotan Kata (*Term Weghting)*

Beberapa hal yang perlu di perhatikan dalam mencari informasi dari lokasi dokumen yang heterogen adalah pembobotan term. Term bias berbentuk kata, frase atau unit hasil dari indexing lainnya dalam suatu dokumen yang digunakan untuk mengetahui konteks dari dokumen tersebut. Setiap kata memiliki tingkat kepentingan yang berbeda dalam dokumen, sehingga setiap kata diberikan sebuah indikator, yaitu *term weight* (Zafikri, 2008).

Menurut Zafikri (2008) *term weighting* atau pembobotan *term* sangat dipengaruhi oleh hal berikut :

1. *Term Frequency (TF)*

   Term frequency adalah faktor untuk menentukan bobot *term* pada suatu dokumen berdasarkan jumlah kemunculan dalam dokumen tersebut. Maka nilai jumlah kemunculan suatu kata (*term frequency)* diperhitungkan dalam pemberian bobot terhadap suatu kata. Jadi semakin besar jumlah kemunculan suatu term (TF tinggi) maka semakin besar pula bobotnya dalam dokumen atau akan memberikan nilai kesesuaian yang semakin besar

2. *Inverse Document Frequency (IDF)*

   Inverse Document Frecuency adalah suatu cara untuk mengurangi dominasi *term* yang sering muncul di seluruh dokumen. Hal ini diperlukan karena *term* yang sering muncul, dapat dianggap sebagai *term* umum (*common term)* sehingga tidak penting nilainya. Sebaliknya, Faktor kejarangan muncul kata *(term scarcity)* dalam koleksi dokumen harus diperhatikan dalam pemberian bobot. ‘Kata yang muncul pada sedikit

dokumen harus dipandang sebagai kata yang lebih penting (*uncomman terms)* daripada kata yang muncul pada banyak dokumen. Pembobotan akan memperhitungkan faktor kebalikan frekuensi dokumen yang menggandung satu kata tersebut.

Metode TF-IDF merupakan metode pembobotan *term* yang sering dijumpai untuk digunakan sebagai metode pembanding terhadap metode pembobotan baru. Pada metode ini, perhitungan bobot *term t* dalam sebuah dokumen dilakukan dengan mengalikan nilai *Term Frequency* dengan *Inverse Document Frequency.*

Pada *Term Frequency* (TF), terdapat jenis formula yang dapat digunakan yaitu

1. *TF biner (binary TF),* yaitu hanya memperhatikan apakah suatu kata ada atau tidak dalam dokumen, jika ada diberi nilai satu dan jika tidak ada maka diberi nilai nol.
2. TF murni (*raw TF),* yaitu nilai TF yang diberikan berdasarkan jumlah kemunculan suatu kata di dokumen. Misalnya, jika suatu kata muncul sebanyak 3 kali maka kata tersebut akan bernilai 3
3. TF logaritmik, yaitu digunakan untuk menghindari dominasi dokumen yang mengandung sedikit kata dalam bentuk *query*, namun mempunyai frekuensi yang tinggi.

$$tf = 1 + \log(tf) \qquad (2.1)$$

4. TF normalisasi, menggunakan perbandingan antara frekuensi sebuah kata dengan jumlah keseluruhan kata pada dokumen

$$tf = 0{,}5 + 0{,}5x \left(\frac{tf}{\max tf}\right) \qquad (2.2)$$

Inverse Document Frequency (idf) diitung dengan menggunakan persamaan berikut :

$$idf_j = log\left(\frac{D}{df_j}\right) \quad (2.3)$$

Keterangan

D : jumlah semua dokumen

idf_j : jumlah dokumen yang mengandung term t_j

Dengan demikian persamaan umum untuk TF-IDF adalah penggabungan dari formula perhitungan *raw* TF dan formula IDF yaitu mengalikan nilai *Term Frequency (TF)* dengan nilai *Inverse Document Frequenncy (idf)* :

$$w_{ij} = tf_{ij}\ x\ idf_j$$

$$w_{ij} = tf_{ij}\ x\ log\left(\frac{D}{df_j}\right) \quad (2.4)$$

Keterangan :

w_{ij} : bobot *term* t_j terhadap dokumen d_i

tf_{ij} : jumlah kemunculan *term* t_j dalam dokumen d_i

2.5 Analisis Sentimen

Analisis sentimen adalah metode untuk menganalisis sebagian data untuk mengetahui emosi manusia. Analisis sentimen dapat dikategorikan ke dalam tiga *task*, yaitu *informative text, detection, information extraction dan sentimen interestiness classification (emotional, polarity identification). Sentimen classification* (negatif dan positif) digunakan untuk memprediksi sentimen *polarity* berdasarkan data sentimen dari pengguna (Liu, 2012).

Secara umum, *opinion mining* diperlukan untuk mengetahui sikap seorang pembicara atau penulis sehubungan dengan beberapa topik atau polaritas kontekstual keseluruhan dokumen. Sikap yang diambil mungkin menjadi pendapat atau penilaian atau evaluasi (teori appraisal), keadaan afektif (keadaan emosional penulis saat menulis) atau komunikasi emosional (efek emosional penulis yang ingin disampaikan pada pembaca) (Saraswati, 2011).

Sentimen Analysis atau *opinion mining* mengacu dalam bidang yang meluas mulai pengolahan Bahasa alami, komputasi *linguistic* dan *text mining* yang tujuannya untuk menganlisa sentimen, pendapat, evaluasi, sikap, penilaian dan emosi seseorang apakah pembicara tersebut atau penulis dapat memberikan suatu topik, produk, layanan, organisasi, individu ataupun kegiatan tertentu (Liu, 2012).

Berdasarkan sumber data, analisis sentimen dibedakan menjadi dua kategori yaitu,

a. *Coarse Grained Sentimen Anlaysis*

   Pada kategori ini, melakukan analisis sentimen pada tingkat dokumen. Dan pada kategori ini juga membuat semua isi dokumen sebagai sebuah sentiment positif dan negatif

b. *Find Grained Sentimen Analysis*

   Kategori ini adalah analisis sentimen pada tingkat kalimat. Pada jenis ini menjadikan setiap kalimat memiliki sentimen yang berbeda meskipun berada dalam satu dokumen. Pada penelitian ini melakukan

analisis sentimen pada level teks, dengan asumsi bahwa status media sosial dapat merupakan bentuk dari sebuah teks yang memiliki sentimen positif dan negatif saja.

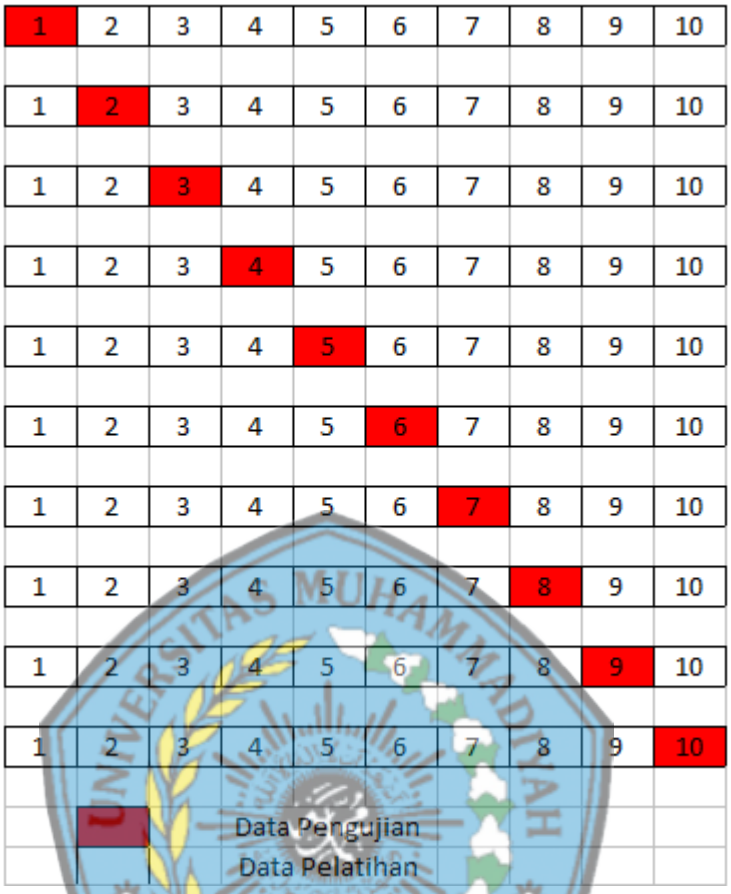
2.6 Klasifikasi

Teknik klasifikasi adalah salah satu teknik data mining yang termasuk *supervised learning*. Dimana, *supervised learning* artinya proses pembentukan sebuah korespondensi menggunakan sebuah *training* dataset. Tujuannya adalah untuk memprediksi target dari beberapa atribut (Zaki & Meira, 2014). Terdapat pada dua pekerjaan utama pada klasifikasi yaitu melakukan *training* untuk disimpan sebagai prediksi dan melakukan *testing* untuk proses klasifikasi agar diketahui di label mana objek data tersebut (Liu, Loh, & Sun, 2009)

2.7 *K-fold Cross Validation*

K-fold cross validation adalah salah satu metode statistik yang digunakan utuk mempartisi data menjadi data *tranning* dan data *testing.* Pada metode ini sering digunakan peneliti karena dapat mengurangi bias yang terjadi dalam pengambilan sampel. Metode ini digunakan secara berulang-ulang membagi data menjadi dua yakni data *tranning* dan data *testing*, setiap data memperoleh kesempatan menjadi data *testing* (Wibowo, 2017).

Selanjutnya pemilihan jenis *cross validation* dapat didasarkan pada ukuran dataset. Biasanya *k-fold cross validation* digunakan karena dapat mengurangi waktu komputasi dengan tetap menjaga keakuratan estimasi



Gambar 2.1 Ilustrasi Pembagian Data

10-*fold cross validation* adalah salah satu *K-fold cross validation* yang direkomendasikan untuk pemilihan model terbaik karena cenderung memberikan estimasi akurasi yang kurang bias dibandingkan dengan *cross validation* biasa, *leave-one-out cross validation* dan *bootstrap*. Dalam 10-*fold cross validation,* data dibagi menjadi 10-*fold* berukuran kira-kira sama, sehingga kita memiliki 10 *subset* data untuk mengevaluasi kinerja model atau algoritma. Untuk masing-masing dari 10 *subset* data tersebut, *cross validation* akan menggunakan 9-*fold* untuk pelatihan (data *training*) dan 1-*fold* untuk pengujian (data *testing*) (Wibowo, 2017).

2.8 Naïve Bayes Classifier

Naïve bayes classifier merupakan Teknik memprediksi dalam mengklasifikasi berbasis probabilistik sederhana berdasarkan pada penerapan teorema *Bayes* (aturan *Bayes*) dengan asumsinya yaitu independensi (ketidaktergantungan) yang kuat (naif). *Naïve Bayes Classifier* mengasumsikan bahwa kehadiran (atau ketiadaan) fitur tertentu dari suatu kelas yang tidak saling berhubungan dengan kehadiran fitur lainnya (Kaku, Mulyanto, & Rohandi, 2014).

Berikut prediksi *Naïve Bayes* didasarkan pada teorema *Bayes* dengan rumus untuk klasifikasi sebagai berikut :

$$P(Y|X) = \frac{P(Y)\ \prod_{i=1}^{q} P(X_i|Y)}{P(X)} \qquad (2.5)$$

Sedangkan *Naïve Bayes* dengan fitur kontnu memiliki formula sebagai berikut

$$P(X|Y) = \frac{1}{\sqrt{2\pi\sigma}} exp^{\frac{-(x-\mu)^2}{2\sigma^2}} \qquad (2.6)$$

Dengan keterangan

P : Parameter

P (Y|X) : Probabilitas data dengan atribut X pada kelas Y (*posterior probability)*

P (Y) : Probabilitas awal kelas Y (*prior probability)*

σ : Standar deviasi

μ : *mean* atau nilai rata-rata dari atribut dengan fitur kontinu

$\prod_{i=1}^{q} P\,(X_i|Y)$: Probabilitas independent kelas Y dari semua fitur dalam

vektor X (*Likelihood)*

Dalam penelitian ini yang digunakan adalah data uji atau status (yang telah diubah menjadi bentuk *term)* yang beredar di media sosial, sedangkan untuk data *tranning* menggunakan responden untuk mengklasifikasikan secara manual sebuah status dan *corpus* (kumpulan kata) berbahasa Indonesia yang telah dipastikan untuk sentimen dan kategorinya yaitu kopi. Dengan menggunakan metode *Naïve Bayes Classifier*, setiap *term* status akan direpresentasikan dengan pasangan atribut "X1,X2,X3, … Xn" dimana X1 adalah *n-gram* pertama, X2 itu *n-gram* kedua dan seterusnya. Sedangkan untuk atribut Y adalah himpunan kategori sentimen. Klasifikasi akan ditentukan dengan mencari probabilitas tertinggi dari semua kategori status yang diujikan *(Vmap)* sehingga diperoleh persamaan berikut :

$$Vmap = \frac{P(X1,X2,X3,\ldots,Xn\,|Yj)P(Yj)}{P(X1,X2,X3,\ldots,Xn)} \quad (2.7)$$

Karena nilai P(X1,X2,X3, … , Xn) bernilai konstan untuk semua kategori (Yj) sehingga persamaan dapat ditulis

$$Vmap = P(X1,X2,X3,\ldots,Xn\,|Yj)P(Yj) \quad (2.8)$$

Persamaan di atas dapat disederhanakan menjadi

$$Vmap = \prod_{i=1}^{n} P(Xi\,|Yj)P(Yj) \quad (2.9)$$

Keterangan :

(Yj) : Kategori klasifikasi atau class

$P(Xi\,|Yj)$: Probabilitas Xi pada kategori Yj

$P(Yj)$: Probabilitas prior dari kategori (Yj)

Untuk P(Yj) dan $P(Xi\ |Yj)$ dihitung pada saat pelatihan dimana persamaan yang diperoleh

$$P(Yj) = \frac{|Yj|}{|Ytotal|}$$

$$P(Xi\ |Yj) = \frac{n_k+1}{n+|kosakata|} \quad (2.10)$$

Keterangan :

$|Yj|$: jumlah dokumen setiap kategori j

$|Ytotal|$: jumlah dokumen dari semua kategori

nk : jumlah frekuensi kemunculaan setiap kata

n : jumlah frekuensi kemunculan kata dari setiap kategori

$|kosakata|$: jumlah semua kata dari semua kategori

2.9 Multinomial Naïve Bayes

Multinomial naïve bayes merupakan proses pengambilan jumlah kata yang muncul dalam setiap dokumen, dimana metode ini mengansumsikan dokumen yang memiliki beberapa kejadian dalam kata dengan panjang tidak bergantung dari kelasnya dalam dokumen tersebut. Menurut , probabilitas pada sebuah dokumen d berada pada kelas c, kondisi ini dapat dinyatakan dengan formula berikut :

$$P(c|d)\ a\ P\ (c) \prod_{1<k<nd} P(t_k|c) \quad (2.11)$$

Untuk rumus $P(t_k|c)$ yakni conditional probabilitas dari kata t_k yang terdapat dalam sebuah dokumen dari kelas c. P(c) merupakan *prior* probabilitas

dari sebuah dokumen yang terdapat dalam kelas c. $(t_1, t_2, \dots t_{nd})$ merupakan token dalam dokumen d yang merupakan bagian dari *vocabulary* yang digunakan sebagai klasifikasi dan merupakan jumlah *token* dalam dokumen d.

Untuk memperkirakan *prior* probabilitas P(c) dapat dilihat formula berikut

$$P\ (c) = \frac{Nc}{N} \qquad (2.12)$$

Keterangan :

Nc : jumlah dokumen *training* dalam kelas c

N : jumlah keseluruhan dokumen *training* dari seluruh kelas

Untuk memperhatikan *conditional probability* P(t|c) dinyatakan dengan rumus berikut :

$$P(t|c) = \frac{Tct}{\sum t' \in V\ Tct} \qquad (2.13)$$

Keterangan

Tct : jumlah kemunculan kata t dalam sebuah dokumen *training* pada kelas c

$\frac{Tct}{\sum t' \in V\ Tct}$: jumlah total keseluruhan kata dalam dokumen *training* pada kelas c

t’ : jumlah total kata dalam dokumen *tranning*

Untuk menghilangkan nilai nol pada sebuah dokumen. Digunakan *laplace smoothing* sebagai proses penambahan nilai 1 pada setiap nilai Tct pada perhitungan *conditional* probabilitas yang dinyatakan dengan formula berikut

$$P(tPtk|c) = \frac{Tct+1}{\sum t' \in V\ Tct+B'} \qquad (2.14)$$

Keterangan

B’ : total kata unik pada keseluruhan kelas dalam dokumen *training*

Untuk mendapatkan nilai probabilitas yang tinggi pada setiap kata maka digunakan *laplace smoothing* atau *add-one*, *laplace smoothing* ini digunakan agar nilai dari probabilitas masing-masing kata dapat memenuhi syarat yaitu tidak sama dengan nol. Jadi, jika suatu nilai pada probabilitas kata adalah nol, maka data tidak baik pada data *training* maupun *testing* juga tidak akan pernah cukup untuk mewakili frekuensi saat terdapat kejadian langkah.

2.10 Word Cloud

Word cloud adalah salah satu hasil dari metode text mining yang menampilkan kata-kata populer terkait dengan kata kunci internet dan data teks. Menurut PBC (dalam Arkhamsiagustinah, 2015) *Word Cloud* sering digunakan untuk menyoroti istilah populer atau trend berdasarkan frekuensi pengguna. Menurut Shawn Graham, Ian Milligan, dan Scott Weingart (dalam Arkhamsiagustinah, 2015) *Word Cloud* merupakan pendekatan yang dapat menjelaskan pertanyaan penelitian dengan sangat cepat dan mudah, dapat menjelajahi word cloud secara singkat dan dapat melakukan analisis yang komprehensif. Kata yang paling sering muncul di dalam data teks akan memiliki bentuk yang paling besar, begitu pula sebaliknya. Berikut adalah contoh dari visual dengan *wordcloud :*

Gambar 2.2 Contoh Visual Kata dengan World Cloud

2.11 Ukuran Evaluasi Model Klasifikasi

Evaluasi pada suatu kalsifikasi pada umunya dilakukan dengan menggunakan sebuah himpunan data yang diuji, tidak digunakan dalam pelatihan klasifikasi tersebut, pada suatu ukuran tertentu. Pada tahap ini terdapat sejumlah ukuran yang dapat digunakan untuk menilai kembali atau mengevaluasi model klasifikasi, yakni *accuracy* atau tingkat pengenalan, *error rate* atau tingkat kesalahan atau kekeliruan klasifikasi, *recall* atau *sensitivity* atau *true positif, specificity* atau *true negative dan precision* (Lim, dkk 2006).

Model klasifikasi yang telah dibuat yakni pemetaan dari suatu baris data dengan keluaran sebuah hasil prediksi kelas/target dari data tersebut. Pada klasifikasi ini memiliki dua kelas sebagai luarannya yang disebut *klasifikasi biner*. Kedua kelas tersebut biasa diinterpretasikan dalam {0,1}, {+1,-1} atau {*positive; negative}* (Lim, dkk 2006).

Dalam proses evaluasi klasifikasi terdapat empat kemungkinan yang terjadi yaitu proses pengklasifikasian pada suatu baris data. Jadi, jika data positif dan diprediksi positif maka akan dihitung sebagai *true positif*, bahkan

jika data itu diprediksi negatif maka akan dihitung sebagai *false negatif*. Jika data negatif dan diprediksi negatif maka akan dihitung sebagai *true negative*, tetapi jika data tersebut diprediksi positif maka akan dihitung sebagai *false positive*. Hasil klasifikasi biner pada suatu dataset yang dipresentasikan dalam bentuk matiks 2 x 2 yakni dinamakan *confusion matrix* (Lim, dkk 2006). Berikut contoh matriks :

Tabel 2.7 Matriks *Contengency* Prediksi dan Aktual

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

Confusion matrix sangat bermanfaat untuk menganalisis kualitas *classfier* dalam mengenali tuple-tuple dari kelas yang ada. TP dan TN menyatakan pada *classifier* mengenali tuple dengan benar, artinya tuple positif dikenali sebagai positif dan tuple negative dikenal sebagai negative. Sedangkan, FP dan FN menyatakan bahwa *classifier* salah dalam mengenali tuple, *tuple negative* dikenali sebagai positif dan tuple negative dikenali sebagai tuple positif. Ada beberapa formula dalam perhtiungan performa klasifikas yaitu nilai akurasi, presisi, dan *recall* biasa ditampilkan dalam presentase

a. *Accurasy*

Akurasi adalah nilai ketepatan dimana, pengguna memprediksi suatu kata sesuai dengan jawaban suatu system (Lim, dkk 2006). Berikut formula nilai akurasi

$$Accuracy = \frac{TP+FN}{TP+FP+TN+FN} \quad (2.15)$$

b. Precission

Precision adalah proposi jumlah dokumen teks yang relevan yang dikenali diantara semua dokumen teks yang dipilih oleh sistem atau kesesuaian terhadap system (Lim, dkk 2006).

$$Precision = \frac{TP}{TP+FP} \quad (2.16)$$

c. *Recall*

Recall adalah proporsi jumlah dokumen teks yang relevan dikenal diantara semua dokumen teks relevan yang ada pada koleksi (Lim, dkk 2006).

$$Recall = \frac{TP}{TP+FN} \quad (2.17)$$