BAB II
TINJAUAN PUSTAKA

Agar penelitian ini lebih terarah, terperinci, dan dapat dipertanggungjawabkan, dibutuhkan kajian pustaka dan dasar teori yang menunjang dengan topik terkait. Untuk itu pada bab ini dibahas mengenai kajian pustaka dan dasar teori dengan topik yang berkaitan dengan penelitian, seperti mengenai buku komputer, *preprocessing*, KNN, *Naive Bayes*, metode gabungan *K-Means* dan LVQ.

2.1 Kajian Pustaka

Dalam International Journal on Computer Science and Engineering (IJCSE), penelitian J. Sreemathy dan P. S. Balamurugan pada tahun 2012 yang berjudul "An Efficient Text Classification Using KNN and *Naive Bayesian*", memaparkan pengkategorian dokumen teks berbahasa Inggris dengan menggunakan metode KNN dan *Naive Bayes*. Dokumen teks yang digunakan dibagi menjadi 2 kelompok, yaitu dokumen yang dibuat sendiri dan dokumen yang diambil dari internet. Dokumen yang dibuat sendiri terdiri atas 150 dokumen dengan 7 kategori, sedangkan dokumen yang diambil dari internet adalah data *reuters21578* yang diambil dari *website* dengan alamat http://kdd.ics.uci.edu/databases/reuters21578/reuters21578. Data dokumen teks tersebut dilakukan proses *text preprocessing*, seleksi fitur, pengurangan vektor fitur, dan diklasifikasikan dengan metode KNN dan *Naive Bayes*. Hasil klasifikasi dievaluasi dengan *recall* dan *precision*. Didapatkan kesimpulan bahwa metode KNN dan *Naive Bayes* lebih efektif dan efisien ketika dibandingkan dengan metode lain, yaitu metode SVM. Selain itu, metode *Naive Bayes* adalah metode yang paling unggul dalam menyelesaikan permasalahan klasifikasi dokumen teks [SRE-12].

Dalam International Journal of Emerging Trends & Technology in Computer Science (IJETICS), penelitian Vaibhav C.Gandhi dan Jignesh A.Prajapati pada tahun 2012 yang berjudul "Review on Comparison between Text Classification Algorithms", memaparkan pengkategorian dokumen teks berbahasa

Inggris dengan menggunakan metode SVM, KNN, dan *Naive Bayesian*. Dokumen teks yang digunakan adalah data *reuters21578* yang diambil dari http://kdd.ics.uci.edu/databases/reuters21578/reuters21578. Data dokumen teks tersebut dilakukan proses *text preprocessing*, dan diklasifikasikan dengan metode SVM, KNN dan *Naive Bayesian*. Hasil penelitian menyatakan bahwa metode SVM dan KNN unggul dibandingkan metode *Naive Bayesian* [GAN-12].

Dalam JURNAL LINK Vol 13/No.1/Januari 2010 , penelitian Cahyo Darujati pada tahun 2010 yang berjudul "Perbandingan Klasifikasi Dokumen Teks Menggunakan Metode *Naïve Bayes* dengan *K-Nearest Neighbor*", memaparkan pengkategorian dokumen teks berbahasa Indonesia dengan menggunakan metode KNN dan *Naive Bayes*. Dokumen teks yang digunakan adalah artikel dari majalah CHIP yang dibagi dalam 5 kelas. Data dokumen teks tersebut dilakukan proses *text preprocessing*, dan diklasifikasikan dengan metode KNN dan *Naive Bayes*. Hasil klasifikasi dievaluasi dengan *recall* dan *precision*. Hasil penelitian menyatakan bahwa metode *Naive Bayes* unggul dibandingkan metode KNN [DAR-10].

Dari tiga penelitian tersebut, terlihat perbedaan hasil penelitian dimana penelitian J. Sreemathy dan P. S. Balamurugan, dan penelitian Cahyo Darujati menyatakan bahwa metode *Naive bayes* adalah metode yang paling unggul dalam menangani klasifikasi dokumen teks, sedangkan penelitian Vaibhav C.Gandhi dan Jignesh A.Prajapati menyatakan bahwa metode SVM dan KNN yang lebih unggul. Karena perbedaan hasil dari beberapa penelitian, belum dapat dihasilkan kesimpulan metode mana yang paling cocok untuk pengkategorian buku komputer berbahasa Indonesia. Untuk itu, penelitian ini melakukan perbandingan terhadap beberapa metode klasifikasi dokumen teks untuk mendapatkan metode yang terbaik dalam menyelesaikan permasalahan pengkategorian buku komputer.

Penelitian Dian Eka Ratnawati, dkk, yang dilakukan di Universitas Brawijaya pada tahun 2012 yang berjudul "Pengembangan Metode Klasifikasi berdasarkan *K-Means* dan LVQ" berhasil mengembangkan metode klasifikasi berdasarkan metode *K-Means* dan metode LVQ. Penelitian ini mengklasifikasikan data kanker payudara yang diambil dari http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/breast-cancer-wisconsin.data, dengan

data latih sejumlah 630 data. Data tersebut dibiarkan apa adanya tanpa dilakukan *preprocessing*. Metode tersebut diujikan pada 50 data uji yang diambil dari sumber yang sama dari data latih. Pengujian dilakukan beberapa kali, berdasarkan parameter dari *K-Means* dan LVQ. Akurasi hasil pengujian cukup tinggi , yaitu sekitar 80%. Hal ini menandakan bahwa metode ini cukup efektif untuk digunakan sebagai metode klasifikasi [RAT-12].

Metode yang dipilih dalam penelitian ini adalah KNN dan *Naïve Bayes*, karena berdasarkan beberapa penelitian di atas, *Naïve Bayes* dan KNN merupakan metode yang cukup efektif dalam menyelesaikan masalah klasifikasi dokumen teks. Metode SVM tidak diambil karena berdasarkan J. Sreemathy dan P. S. Balamurugan, hasil klasifikasi metode ini sangat buruk. Selain kedua metode tersebut, metode gabungan *K-Means* dan LVQ juga dipilih karena metode ini sebelumnya belum pernah dibandingkan dengan metode KNN dan *Naïve Bayes*, sehingga dapat diuji keefektifan metode ini terhadap pengkategorian buku komputer berbahasa Indonesia.

2.2 Dasar Teori

Dasar teori yang terkait dengan penelitian ini antara lain dasar teori mengenai buku komputer; konsep umum klasifikasi buku yang terdiri dari klasifikasi buku, tujuan klasifikasi buku, manfaat klasifikasi buku, dan cara penentuan klasifikasi Buku; *teks mining; text preprocessing* yang terdiri dari *case folding, tokenizing, filtering*, dan *stemming*; pembobotan Tf-Idf; *Naive Bayes, k-Nearest Neighbor, Cosine Similarity, Learning Vector Quantization* (LVQ), *k-Means, euclidean distance*, metode gabungan *k-Means* dan LVQ; dan akurasi sistem.

2.2.1. Buku Komputer

Objek yang diteliti dalam penelitian ini adalah buku komputer berbahasa Indonesia. Untuk lebih memahami objek diteliti, alangkah baiknya jika dikenali definisi dari buku, komputer, dan buku komputer. Menurut Kamus Besar Bahasa Indonesia, buku adalah lembar kertas yg berjilid, dapat berisi tulisan atau kosong [DEP-08]. Sedangkan menurut Jeremias Jena, buku adalah susunan atau

kumpulan/gabungan kertas-kertas dalam ukuran tertentu yang salah satu fungsinya sebagai bentuk penyimpanan data / pengetahuan / sejarah suatu bangsa serta sebagai sumber referensi yang dibutuhkan banyak kalangan [JEN-08]. Dengan demikian, dapat disimpulkan bahwa buku merupakan kumpulan dari kertas yang dapat mengandung informasi maupun tidak yang dijilid / digabung.

Definisi Komputer sangat luas dan beragam. Robert H. Bissmer mengungkapkan bahwa komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas, yaitu menerima *input*, memproses *input*, menyimpan perintah-perintah dan hasil pengolahannya, dan menyediakan *output* dalam bentuk informasi [JOG–99]. Sedangkan Donald H. Sanders mengungkapkan bahwa komputer adalah sistem elektronik untuk memanipulasi data yang cepat dan tepat serta dirancang dan diorganisasikan supaya secara otomatis menerima dan menyimpan data *input*, memprosesnya, dan menghasilkan *output* dibawah pengawasan suatu langkah-langkah instruksi-instruksi program yang tersimpan di *memori* [JOG-99]. Menurut William M. Fuori, komputer adalah suatu pemroses data yang dapat melakukan perhitungan yang besar dan cepat, tanpa campur tangan dari manusia selama pemrosesan berlangsung [JOG-99]. Dari beberapa definisi yang diungkap para ahli, dapat disimpulkan bahwa komputer adalah alat elektronik yang dirancang untuk menerima data masukan, mengolah data tersebut dengan suatu instruksi tertentu, dan memberi informasi keluaran dimana proses pengolahan berlangsung secara otomatis / tanpa campur tangan dari manusia.

Dari definisi buku dan komputer, dapat disimpulkan bahwa buku komputer adalah kumpulan kertas yang mengandung informasi seputar komputer dan segala bidang yang berkaitan dengan komputer. Dengan kata lain, Buku komputer merupakan buku yang membahas segala sesuatu yang ada dalam ruang lingkup komputer, seperti teknologi perangkat keras komputer, pemrosesan dalam komputer, jaringan komputer, dan lain sebagainya.

2.2.2 Konsep Umum Klasifikasi Buku

Hal yang dilakukan dalam penelitian ini adalah pengkategorian atau yang lebih dikenal dengan istilah klasifikasi. Dalam konsep umum klasifikasi ini dibahas mengenai definisi klasifikasi, tujuan klasifikasi buku, manfaat klasifikasi buku, dan Cara Penentuan Klasifikasi Buku.

2.2.2.1 Klasifikasi Buku

Untuk mengetahui hal yang akan dilakukan dalam penelitian ini, yaitu klasifikasi, alangkah baiknya jika dipahami pengertian klasifikasi terlebih dahulu. Menurut Kamus Besar Bahasa Indonesia, klasifikasi adalah penggolongan (menurut jenis); penyusunan dalam golongan-golongan; pembagian menjadi golongan-golongan [DEP-08]. Menurut Sulistyo-Basuki, klasifikasi yang berasal dari bahasa Latin yaitu *classis* yang berarti kelas adalah proses pengelompokan artinya mengumpulkan benda/entitas yang sama serta memisahkan benda/entitas yang tidak sama [SUL-91]. Menurut Yusup bagan atau skema klasifikasi didefinisikan sebagai suatu susunan kelompok kelas yang kemudian dibagi ke dalam golongan-golongan yang mempunyai sifat dan ciri yang sama [HAM-95]. Dari beberapa definsi di atas, dapat didefinisikan bahwa klasifikasi merupakan kegiatan mengelompokan benda / entitas ke dalam golongan berdasarkan kesamaan sifat / cirinya.

Buku termasuk salah satu bentuk dokumen teks, sehingga buku dapat diklasifikasikan dengan menggunakan pendekatan klasifikasi dokumen teks. Menurut Manning, dkk, klasifikasi dokumen merupakan proses menemukan sekumpulan model yang mendeskripsikan dan membedakan kelas-kelas dokumen sesuai dengan kategori yang dimilikinya [MAN-08]. Tujuan klasifikasi dokumen teks adalah memprediksikan kelas dari dokumen teks yang belum diketahui kelasnya dengan karakteristik tipe data yang bersifat kategorik.

Proses klasifikasi dibagi menjadi dua fase, yaitu *learning* dan *test*. Pada fase *learning*, sebagian data yang telah diketahui kelas datanya (*training set*) digunakan untuk membentuk model. Selanjutnya pada fase *test*, model yang sudah terbentuk diuji dengan sebagian data lainnya untuk mengetahui akurasi model

tersebut. Jika akurasinya mencukupi maka model tersebut dapat dipakai untuk memprediksi kelas data yang belum diketahui.

Metode klasifikasi dokumen secara otomatis memiliki tingkat keakuratan yang cukup tinggi jika aturan dibuat dengan baik. Menurut Manning, dkk klasifikasi dokumen secara otomatis terdiri atas dua kategori, yaitu *hand-coded rule-based systems* dan *supervised learning* [MAN-08]. *Hand-coded rule-based systems* adalah metode klasifikasi dengan menggunakan sekumpulan aturan yang terdapat di dalam basis pengetahuan dan dengan menggunakan mesin inferensi untuk menghasilkan informasi baru. *Hand-coded rule-based systems* tingkat akurasinya tinggi jika aturan dibuat dengan sangat baik dan kompleks serta butuh biaya yang mahal. *Supervised learning* menggunakan data latih untuk memberikan label kategori yang telah terdefinisi sebelumnya. Beberapa model yang digunakan dalam *supervised learning* adalah *naive bayes, k-nearest neighbors, support vector machine* , dan lain sebagainya.

Dalam penelitian ini digunakan klasifikasi dokumen teks kategori *supervised learning*, dengan metode *k-nearest neighbors* (KNN), *naive bayes*, dan metode baru yang dikembangkan oleh Dian Eka Ratnawati, dkk pada tahun 2012, yaitu metode gabungan *K-Means* dan *LVQ* [RAT-12].

2.2.2.2 Tujuan Klasifikasi Buku

Dalam penelitian ini, dilakukan klasifikasi terhadap buku komputer. Para ahli telah mengungkap beberapa tujuan klasifikasi buku, khusunya tujuan klasifikasi buku dalam perpustakaan. Menurut Sundari klasifikasi mempunyai lima tujuan yaitu [HAM-95] :

1. Untuk menetapkan dan menunjukkan isi pokok yang dibahas dalam suatu bahan pustaka.
2. Untuk mengumpulkan bahan pustaka yang bidang kajian atau subjeknya sama ke dalam suatu kelompok subjek tertentu.
3. Untuk memudahkan dan memandu pengguna atau pustakawan dalam mencari dan menemukan kembali koleksi atau sekumpulan koleksi pustaka bilamana diperlukan.

4. Untuk menentukan letak dan susunan koleksi pustaka dalam rak dan kartu katalog subjek dalam laci katalog.
5. Memandu pengguna menemukan sekumpulan dokumen dalam subjek yang berkaitan (relevan) satu sama lain sewaktu mereka melakukan pencarian sendiri ke koleksi (browsing).

Sulistyo-Basuki memaparkan tujuan klasifikasi buku dalam perpustakaan, yaitu [SUL-91] :

1. Menghasilkan urutan yang bermanfaat.
2. Penempatan yang tepat.
3. Penyusunan mekanis.
4. Tambahan dokumen baru, sehingga klasifikasi perpustakaan harus mampu menentukan lokasi yang paling bermanfaat bagi dokumen baru di antara dokumen lama.
5. Penarikan dokumen dari rak.

Dari kedua pendapat di atas dapat dilihat bahwa tujuan klasifikasi buku adalah untuk menyusun buku yang dimiliki perpustakaan sedemikian rupa dengan tidak memandang besar kecilnya koleksi perpustakaan, sehingga memudahkan dalam pencarian dan peletakan buku.

2.2.2.3 Manfaat Klasifikasi Buku

Klasifikasi buku tentu memiliki sejumlah manfaat, khususnya bagi perpustakaan. Menurut Eryono, manfaat klasifikasi antara lain [ERY-93] :

1. Buku-buku yang sama atau mirip isinya akan terletak berdekatan.
2. Memudahkan dalam mengadakan perimbangan koleksi yang dimiliki.
3. Memudahkan dalam mengadakan penelusuran terhadap bahan pustaka menurut subjek.
4. Memudahkan dalam pembuatan bibliografi (cara untuk memberitahukan adanya suatu buku/pustaka dan sejumlah buku/pustaka yang pernah diterbitkan) menurut pokok masalah.

Dari pendapat di atas dapat diketahui bahwa klasifikasi perpustakaan banyak memberikan kemudahan baik bagi pustakawan maupun pengguna

perpustakaan, yaitu memudahkan pencarian, peletakan, dan pembuatan bibliografi dari sebuah bahan pustaka.

2.2.2.4 Cara Penentuan Klasifikasi Buku

Sebelum menempatkan suatu koleksi (misalnya buku) pada kelas atau golongan yang sesuai, pertama-tama perlu diketahui terlebih dahulu tema atau subjek apa yang dibahas pada buku tersebut. Menurut Towa P. Hamakonda, tema atau subjek suatu buku dapat diketahui dengan cara sebagai berikut [HAM-95] :

1. Dari judul buku.

   Judul suatu buku adakalanya sudah menunjukkan tema atau subjek yang dibahas. Misalnya: Pengantar Aljabar untuk Perguruan Tinggi maka subjeknya adalah "Aljabar", Pengantar Teknologi Informatika maka subjeknya adalah "Informatika", dan seterusnya.
2. Dari sinopsis / ringkasan / resume buku.

   Apabila dari judul suatu buku tidak dapat diketahui secara pasti tema atau subjeknya, maka ringkasan isi buku yang umumnya tertera pada cover belakang dapat dijadikan pedoman. Dari ringkasan itu akan diketahui subjek yang terkandung pada buku tersebut. Misalnya: Layar Terkembang, judul tersebut tidak langsung menunjukkan subjek yang dimaksud tapi harus terlebih dahulu dibaca ringkasannya agar diketahui subjek apa yang dikandungnya, kemudian ditemukan subjek dari buku Layar Terkembang adalah tentang fiksi.
3. Dari daftar isi atau kata pengantar atau pendahuluan buku.

   Apabila dari judul buku belum dapat diketahui, maka subjek yang dibahas dalam buku tersebut dapat ditemukan dengan cara menganalisa daftar isi atau kata pengantar atau pendahuluan buku tersebut.
4. Namun apabila ketiga cara di atas belum memadai untuk menentukan tema atau subjek suatu buku, maka cara terakhir adalah dengan membaca sebagian isi dari buku tersebut.

Dari uraian di atas, variabel terpenting dalam menentukan nomor klasifikasi buku adalah judul, kemudian sinopsis / ringkasan / resume, daftar isi / kata pengantar / pendahuluan, barulah isi dari buku. Penelitian ini membatasi

penggunaan variabel sebatas judul dan sinopsis mengingat kedua variable ini adalah variabel terpenting dalam penentuan nomor klasifikasi buku. Menurut Arif Jacob, judul buku adalah nama yang dipakai untuk buku yang merupakan identitas atau cermin dari jiwa seluruh isi buku, bersifat menjelaskan diri dan yang menarik perhatian [JAC-10]. Menurut Kamus Besar Bahasa Indonesia, sinopsis adalah ikhtisar karangan yang biasanya diterbitkan bersama-sama dengan karangan asli yg menjadi dasar sinopsis itu; ringkasan; abstraksi [DEP-08].

2.2.3 *Text Mining*

Untuk dapat mengklasifikasikan buku komputer berbahasa Indonesia, penelitian ini menerapkan ilmu *text mining.* Menurut Ronen Fieldman, *text mining* dapat didefinisikan sebagai proses mendapatkan informasi secara intensif dimana pengguna berinteraksi dengan koleksi-koleksi dokumen menggunakan seperangkat tools analisis [FIE-07]. Secara umum, proses-proses pada text mining adalah mengadopsi dari proses data mining. Oleh karena itu, *text mining* dan data mining mempunyai banyak kesamaan arsitekturnya. Proses-proses yang ada pada *text mining* juga hampir sama dengan data mining. Proses-proses utama pada *text mining* diantaranya pemrosesan awal (*text preprocessing*), penemuan pola (*pattern discovery*), transformasi teks (*text transformation*), dan pemilihan fitur (*feature selection*).

2.2.4 *Text Preprocessing*

Agar dapat menghasilkan data yang siap diolah ke dalam metode klasifikasi teks, data teks perlu dilakukan *text preprocessimg.* Fadillah Z. Tala mengungkapkan beberapa teori mengenai *text preprocessing. Text preprocessing* adalah tahapan untuk mempersiapkan teks menjadi data yang akan diolah di tahapan berikutnya. Inputan awal pada proses ini adalah berupa dokumen teks. Teks yang akan dilakukan proses text mining pada umumnya memiliki beberapa karakteristik, diantaranya adalah memiliki dimensi yang tinggi, terdapat *noise* pada data, dan terdapat struktur teks yang tidak baik. Agar dapat dihasilkan fitur yang baik dan mewakili data dengan baik, perlu dilakukan tahapan *preprocessing*. *Text preprocessing* pada penelitian ini terdiri dari beberapa tahapan, yaitu: proses

case folding, proses *tokenizing* kata, proses *filtering*, dan proses *stemming* [TAL-03].

2.2.4.1 Case Folding

Case folding adalah tahapan proses mengubah semua huruf dalam teks dokumen menjadi huruf kecil, serta menghilangkan karakter selain a-z, kecuali karakter pemecah kalimat, seperti spasi, tab, dan *newline* (lompat baris) [TAL-03].

2.2.4.2 Tokenizing

Tokenizing adalah proses pemotongan string input berdasarkan tiap kata yang menyusunnya. Pemecahan kalimat menjadi kata-kata tunggal dilakukan dengan melihat pemisah seperti spasi, tab, dan *newline* (lompat baris) [TAL-03].

2.2.4.3 Filtering

Filtering merupakan proses penghilangan *stopword*. *Stopword* adalah kata-kata yang sering kali muncul dalam dokumen namun artinya tidak deskriptif dan tidak memiliki keterkaitan dengan tema tertentu. Didalam bahasa Indonesia *stopword* dapat disebut sebagai kata tidak penting [TAL-03].

2.2.4.4 Stemming

Stemming merupakan proses mencari akar (*root)* kata dari tiap *token* kata yaitu dengan pengembalian suatu kata berimbuhan ke bentuk dasarnya (*stem*). Penelitian ini menggunakan algoritma *porter* untuk kata berbahasa Indonesia. Algoritma *Porter* adalah cara pencarian *root word* (kata dasar) yang dilakukan secara stripping imbuhan dan akhiran tanpa memperhatikan sisipan dan tanpa pengecekan kamus kata dasar. *Porter Stemmer* untuk bahasa indonesia dikembangkan oleh Fadillah Z. Tala pada tahun 2003 yang diimplementasikan berdasarkan *English Portyer Stemmer* yang dikembangkan oleh W. B. Frakes pada tahun 1992. Adapun algoritma *Porter* yang digunakan untuk bahasa Indonesia adalah [TAL-03] :

1. Hapus partikel infleksional.

2. Hapus possessive pronoun (kata ganti kepunyaan).
3. Jika ada awalan pertama, hapus awalan pertama dan lanjut ke langkah 4b, Namun jika tidak ada, lanjut ke langkah 4a.
4. a. Hapus awalan kedua dan lanjut ke langkah 5a.
   b. Jika ada akhiran, hapus akhiran dan lanjut ke langkah 5b. Namun jika tidak ada, kata diasumsikan sudah berupa kata dasar.
5. a. hapus akhiran. Kata yang dihasilkan sudah berupa kata dasar.
   b. Hapus awalan kedua. Kata yang dihasilkan sudah berupa kata dasar.

Terdapat lima aturan pada proses *stemming* untuk Bahasa Indonesia, yaitu :

1. Aturan terhadap partikel infleksional / *particle infleksional*

Tabel 2.1 Aturan Terhadap Partikel Infleksional

Akhiran	Pengganti	Syarat khusus	Contoh
-lah	-	-	pergilah
-kah	-	-	adakah
-tah	-	-	siapatah
-pun	-	-	apapun

2. Aturan terhadap kata ganti kepunyaan / *possessive pronoun*

Tabel 2.2 Aturan Terhadap Kata Ganti Kepunyaan

Akhiran	Pengganti	Syarat Khusus	Contoh
-ku	-	-	bukuku
-mu	-	-	mejamu
-nya	-	-	tasnya

3. Aturan terhadap awalan pertama / *first prefix*

Tabel 2.3 Aturan Terhadap Awalan Pertama

Awalan	Pengganti	Syarat Khusus	Contoh
meng-	-	-	mengambil
meny-	s	kata kerja	menyapu
men-	-	-	menduduki
mem-	p	kata kerja	memaksa

mem-	-	-	mempunyai
me-	-	-	melarang
peng-	-	-	penggusuran
peny-	s	kata kerja	penyaksian
pen-	-	-	penjajah
pem-	p	kata kerja	pemahat
pem-	-	-	pembantu
di-	-	-	ditanya
ter-	-	-	terjadi
ke-	-	-	keluar

4. Aturan terhadap awalan kedua / *second prefix*

Tabel 2.4 Aturan Terhadap Awalan Kedua

Awalan	Pengganti	Syarat Khusus	Contoh
ber-	-	-	bersalah
bel-	-	diikuti ajar	belajar
be-	-	diikuti k	bekerja
per-	-	-	perjelas
pel-	-	diikuti ajar	pelajar
pe-	-	-	pekerja

5. Aturan terhadap akhiran / *derivation suffix*

Tabel 2.5 Aturan Terhadap Akhiran

akhiran	Pengganti	Syarat Khusus	Contoh
-kan	-	prefix bukan anggota ke-, peng-	tarikkan
-an	-	prefix bukan anggota di-, meng-,ter-	makanan
-i	-	prefix bukan anggota ber-, ke-, peng-	tandai

2.2.5 Pembobotan *TF-IDF*

Menurut Grossman, pembobotan dapat diperoleh berdasarkan jumlah kemunculan suatu *term* (kata) dalam sebuah dokumen *term frequency* (*tf*) dan jumlah kemunculan *term* dalam koleksi dokumen *inverse document frequency* (*idf*). *Term frequency* (*tf*) adalah frekuensi kemuculan term pada dokumen. *Document frequency* (*df*) adalah banyaknya dokumen dimana suatu *term* (*t*) muncul. *Inverse document frequency* (*idf*) adalah nilai inverse dari *Document frequency* (*df*). Bobot suatu kata semakin besar jika kata tersebut sering muncul dalam suatu dokumen dan semakin kecil jika kata tersebut muncul dalam banyak dokumen. Nilai *idf* sebuah *term* dapat dihitung menggunakan persamaan sebagai berikut [GRO-98] :

$$IDF = \log\left(\frac{D}{dfi}\right) \quad (2\text{-}1)$$

D adalah jumlah dokumen dan *dfi* adalah jumlah kemunculan (frekuensi) *term* terhadap *D*.

Adapun persamaan yang digunakan untuk menghitung bobot (*W*) masing-masing dokumen terhadap kata kunci (*query*), yaitu:

$$W_{d,t} = tf_{d,t} * IDF_t \quad (2\text{-}2)$$

dengan keterangan sebagai berikut :

d = dokumen ke–d

t = *term* ke–t dari kata kunci

tf = *term* frekuensi/frekuensi kata

$W_{d,t}$ = bobot dokumen ke–d terhadap term ke–t.

2.2.6 *Naive Bayes*

Salah satu metode klasifikasi yang digunakan dalam penelitian ini adalah metode *Naive Bayes* atau yang juga biasa disebut *Naive Bayes classifier* (NBC). Menurut Manning Cd, dkk, *Naive Bayes classifier* merupakan penyederhanaan dari *bayesian classification* dan biasa disebut *simple bayesian classification*. Klasifikasi NBC termasuk dalam *multinomial* yang mengambil jumlah kata yang muncul pada sebuah dokumen. Pada model ini, sebuah dokumen terdiri atas beberapa kejadian kata dan diasumsikan panjang dokumen tidak bergantung pada

kelasnya. Lainnya adalah kemungkinan tiap kejadian kata dalam sebuah dokumen adalah bebas dan tidak terpengaruh dengan konteks kata atau posisi kata dalam dokumen. NBC merupakan metode klasifikasi dengan cara menghitung peluang sebuah dokumen *d* berada di kelas *c* [MAN-08].

Menurut McCallum, Teorema Bayes sendiri berawal dari rumus [MCG-98] :

$$P(A|B) = \frac{P(B \cap A)}{P(B)} \quad (2\text{-}3)$$

dimana *P*(*A*|*B*) adalah peluang keadaan *A* jika diketahui keadaan *B* dan *P(B∩A)* adalah peluang keadaan *B* dan keadaan *A*. *P(B∩A)* dapat diuraikan menjadi :

$$P(B \cap A) = P(B|A)P(A) \quad (2\text{-}4)$$

dengan *P(B|A)* berarti peluang keadaan *B* jika diketahui keadaan *A* dan *P(A)* adalah peluang keadaan *A*. Dari kedua persamaan tersebut, didapatkan teorema *Bayes*, yaitu :

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (2\text{-}5)$$

Menggunakan teorema *Bayes* ini, persamaan diatas dapat ditulis menjadi:

$$V_{MAP} = \underset{\text{vj}\in\text{V}}{\arg\max} \frac{P(a_1,a_2,a_3,\ldots,a_n|V_j)P(V_j)}{P(a_1,a_2,a_3,\ldots,a_n)} \quad (2\text{-}6)$$

dimana V_{MAP} adalah kelas dengan peluang terbesar, $P(a_1, a_2, a_3, \ldots, a_n|V_j)$ adalah peluang keadaan a_1,a_2,a_3, hingga a_n dengan syarat masuk dalam kelas V_j, $P(V_j)$ adalah peluang kelas V_j, dan $P(a_1, a_2, a_3, \ldots, a_n)$ adalah peluang keadaan a_1,a_2,a_3, hingga a_n. Karena nilai $P(a_1, a_2, a_3, \ldots, a_n)$ konstan untuk semua v_j, maka persamaan ini dapat ditulis menjadi:

$$V_{MAP} = \underset{\text{vj}\in\text{V}}{\arg\max} P(a_1, a_2, a_3, \ldots, a_n|V_j)P(V_j) \quad (2\text{-}7)$$

Naïve Bayesian Classifier menyederhanakan hal ini dengan asumsi bahwa fitur-fitur yang terdapat didalamnya saling tidak tergantung atau independen, setiap kata independen satu sama lain. Dengan kata lain :

$$P(a_1, a_2, a_3, \ldots, a_n|V_j) = \prod_i P(a_i|V_j) \quad (2\text{-}8)$$

Dengan men-substitusikan persamaan ini dengan persamaan diatas akan menghasilkan:

$$V_{MAP} = \underset{\text{vj}\in\text{V}}{\arg\max} P(V_j) \prod_i P(a_i|V_j) \quad (2\text{-}9)$$

McCallum, memaparkan penerapan naive bayes classifier dalam klasifikasi dokumen teks. Nilai *P(vj)* ditentukan pada saat pelatihan, yang nilainya didekati dengan :

$$P(V_j) = \frac{|doc_j|}{|contoh|} \quad (2\text{-}10)$$

dimana $P(V_j)$ adalah peluang kemunculan dokumen yang memiliki kategori j, $|doc_j|$ adalah banyaknya dokumen yang memiliki kategori j dalam pelatihan, sedangkan $|contoh|$ banyaknya dokumen dalam contoh yang digunakan untuk pelatihan.

Untuk $P(a_i|V_j)$, yaitu probabilitas kata a_i dalam kategori j ditentukan dengan :

$$P(a_i|V_j) = \frac{n_i + 1}{n + |vocabulary|} \quad (2\text{-}11)$$

dimana n_i adalah frekuensi munculnya kata a_i dalam dokumen yang berkategori *Vj*, sedangkan nilai *n* adalah jumlah kata dalam seluruh dokumen, dan |v*ocabulary*| adalah banyaknya kata unik dalam contoh pelatihan.

Berdasarkan uraian di atas, dapat diketahui algoritma *Naive Bayes Classifier*, yaitu [MCC-98] :

A. Proses pelatihan. Input adalah dokumen latih.

1. Bentuk *token-token* (kosakata) dari dokumen latih.
2. Untuk setiap kategori V_j lakukan :
   a. Hitung |*Docj*| (himpunan dokumen yang berada pada kategori vj).
   b. Hitung $P(V_j)$ dengan persamaan 2-10.
   c. Untuk setiap kata a_i pada kosakata lakukan :
      - Hitung $P(a_i|V_j)$ dengan persamaan 2-11.

B. Proses klasifikasi. Input adalah dokumen uji.

Hasilkan V_{MAP} sesuai dengan persamaan 2-9 dengan menggunakan $P(V_j)$ dan $P(a_i|V_j)$ yang telah diperoleh dari pelatihan.

2.2.7 *K-Nearest Neighbor* (KNN)

Salah satu metode klasifikasi yang digunakan dalam penelitian ini adalah metode *K-Nearest Neighbor* (KNN). Berdasarkan Santosa, *K-Nearest Neighbor* (KNN) adalah sebuah algoritma yang cukup populer untuk melakukan

pengkategorian teks dan merupakan salah satu metode terbaik dalam bidang tersebut. KNN melakukan klasifikasi terhadap obyek berdasarkan data latih yang jaraknya paling dekat dengan obyek tersebut. Teknik ini sangat sederhana dan mudah diimplementasikan. Dalam hal ini jumlah data/tetangga terdekat ditentukan oleh *user* yang dinyatakan dengan *k*. Misalkan ditentukan *k*=5, maka setiap *data testing* dihitung jaraknya terhadap *data training* dan dipilih 5 *data training* yang jaraknya paling dekat ke *data testing*. Lalu periksa *output* atau labelnya masing-masing, kemudian tentukan *output* mana yang frekuensinya paling banyak. Lalu masukkan suatu *data testing* ke kelompok dengan *output* paling banyak. Misalkan dalam kasus klasifikasi dengan 3 kelas, lima data tadi terbagi atas tiga data dengan *output* kelas 1, satu data dengan *output* kelas 2 dan satu data dengan *output* kelas 3, maka dapat disimpulkan bahwa *output* dengan label kelas 1 adalah yang paling banyak. Maka data baru tadi dapat dikelompokkan ke dalam kelas 1. Prosedur ini dilakukan untuk semua *data testing* [SAN-07].

Algoritma KNN adalah [SAN-07] :

1. Input berupa data latih, data uji dan nilai *k*.
2. Hitung jarak data uji dengan setiap data latih.
3. Urutkan data latih berdasarkan kedekatan jarak.
4. Ambil sejumlah k data latih teratas.
5. Tentukan kelas klasifikasi berdasarkan kelas yang paling dominan dari *k* data latih yang telah diambil.

Menurut Anna Huang, untuk mendefinisikan jarak antara dua titik dalam metode KNN yaitu titik pada data teks uji dan titik pada data teks latih dapat digunakan berbagai macam metode, seperti *euclidean distance*, *cosine similarity*, *jaccard coefficient*, p*earson correlation coefficient*, maupun *averaged kullback-leibler divergence* [HUA-08]. Penelitian ini menggunakan *cosine similarity* sebagai metode penentuan jarak titik uji dan titik latih untuk metode KNN ini.

2.2.8 *Cosine Similarity*

Menurut Grossman, *Cosine similarity* digunakan untuk menghitung pendekatan relevansi *query* terhadap dokumen. Penentuan relevansi sebuah *query*

terhadap suatu dokumen dipandang sebagai pengukuran kesamaan antara vektor *query* dengan vektor dokumen. Semakin besar nilai kesamaan vektor *query* dengan vektor dokumen maka *query* tersebut dipandang semakin relevan dengan dokumen. Pada umumnya *cosine similarity (COS*) dihitung dengan rumus *cosine measure* berikut [GRO-98] :

$$\cos(b1, b2) = \frac{\sum_{t=1}^{n} W_{t,b1} W_{t,b2}}{\sqrt{\sum_{t=1}^{n} W^2_{t,b1} \sum_{j=1}^{n} W^2_{t,b2}}} \quad (2\text{-}12)$$

dengan :

$\cos(b1, b2)$ = *cosine similarity* dokumen b1 dan dokumen b2.

t = *term* ke-t dalam kalimat.

$W_{t,b1}$ = bobot term *t* dalam dokumen b_1

$W_{t,b2}$ = bobot term *t* dalam dokumen b_2.

2.2.9 *Learning Vector Quantization* (LVQ)

Learning Vector Quantization (LVQ) adalah suatu metode untuk melakukan pembelajaran pada lapisan kompetitif yang terawasi. Suatu lapisan kompetitif akan secara otomatis belajar untuk mengklasifikasikan vektor-vektor input. Jika 2 vektor input mendekati sama, maka lapisan kompetitif akan meletakkan kedua vektor tersebut ke dalam kelas yang sama

Pada proses awal pengenalan, vektor input akan mengalami proses pembelajaran yang dilakukan melalui beberapa *epoh* sampai batas *epoh* (*MaxEpoh*) maksimal tercapai. Parameter-parameter yang digunakan pada metode LVQ ini adalah sebagai berikut [MAR-99] :

1. *Alfa* / α (*learning rate*) yaitu tingkat pembelajaran. Jika α terlalu besar, maka algoritma akan menjadi tidak stabil sebaliknya jika α terlalu kecil, maka prosesnya akan terlalu lama. Nilai alfa adalah $0 < \alpha < 1$.
2. *DecAlfa / Decα* (penurunan *learning rate*) yaitu penurunan tingkat pembelajaran.
3. *MinAlfa / Minα* (minimum *learning rate / error minimum*) yaitu minimal nilai tingkat pembelajaran yang masih diperbolehkan.

4. *MaxEpoh* (maksimum *epoh*) yaitu jumlah *epoh* atau iterasi maksimum yang boleh dilakukan selama pelatihan. Iterasi akan dihentikan jika nilai *epoh* melebihi maksimum *epoh*.

Berdasarkan Sri Kusumadewi, algoritma LVQ adalah [KUS-03]:

1. Tetapkan :
   a. Bobot awal variabel untuk setiap input ke-*j* menuju ke kelas ke-*i : Wij*.
   b. Maksimum *epoh* : *MaxEpoh*.
   c. Parameter *learning rate* : α.
   d. Pengurangan *learning rate* : *Decα*.
   e. Minimum *learning rate* yang diperbolehkan: *Minα*.
2. Masukkan :
   a. Data input untuk setiap input ke-*j* menuju ke kelas ke-*i* : X_{ij};
   b. Target berupa kelas: T_i;
3. Tetapkan kondisi awal: *epoh* = 0;
4. Kerjakan jika: (*epoh* ≤ *MaxEpoh*) dan (α ≥ *Minα*)
   a. *epoh* = *epoh*+1;
   b. Kerjakan untuk *i*=1 sampai *n*
      i. Tentukan j sedemikian hingga $\|X_i - W_j\|$ minimum (pada penelitian ini digunakan metode euclidean distance) ;
      ii. Perbaiki *Wj* dengan ketentuan :
      Jika *T = Cj* (kelas dari bobot) maka : *Wj = Wj + α (Xi - Wj)*
      Jika *T ≠ Cj* maka : *Wj = Wj - α (Xi - Wj)*
   c. Kurangi nilai α (α = α - *Decα*)

2.2.10 *K-means*

K-Means termasuk *clustering* (pengelompokan) yang memisahkan data ke *k* daerah bagian yang terpisah. *K-Means* sangat terkenal karena kemudahan dan kemampuannya untuk mengklaster data besar dan dengan sangat cepat. Dalam metode *K-Means*, setiap data harus masuk *cluster* (kelompok) tertentu. Kelemahan metode ini memungkinkan bagi setiap data yang termasuk *cluster* tertentu pada suatu tahapan proses, pada tahapan berikutnya berpindah ke *cluster* yang lain. Prinsip utama dari teknik ini adalah menyusun *k* buah prototipe / pusat

massa (*centroid*) / rata-rata (*mean*) dari sekumpulan data berdimensi *n*. Teknik ini mensyaratkan nilai *k* sudah diketahui sebelumnya [LAR-05].

Secara lebih detil algoritma *K-Means* adalah seperti berikut [LAR-05] :

1. Tentukan *k* (jumlah *cluster*) yang ingin dibentuk.
2. Bangkitkan *k* centroid (titik pusat *cluster*) awal secara random.
3. Untuk setiap *record* , temukan pusat *cluster* terdekat dengan mengukur jarak *record* dengan pusat *cluster*. Metode perhitungan jarak yang digunakan adalah *euclidean distance.*
4. Untuk setiap *k cluster* , temukan pusat *cluster* ,dan ubah lokasi dari setiap pusat *cluster* dengan nilai *centroid* yang baru. Pusat *cluster* diperoleh dengan cara menghitung nilai rata-rata dari data-data yang berada pada *cluster* yang sama.
5. Kembali ke langkah 3 – 5 sampai *konvergen.*

Karakteristik dari K-Means [LAR-05] :

1. *K- Means* sangat cepat dalam proses *clustering*.
2. *K- Means* sangat sensitif pada pembangkitan *centroid* awal secara random.
3. Memungkinkan suatu *cluster* tidak mempunyai anggota.
4. Hasil *clustering* dengan *K-Means* selalu berubah-ubah, kadang baik, kadang jelek.
5. *K- Means* sangat sulit untuk mendapat optimum global.

2.2.11 *Euclidean Distance*

Pada penelitian ini, untuk mengukur jarak data ke bobot pada metode LVQ dan mengukur jarak data ke pusat *cluster* pada metode *k-Means* digunakan metode *euclidean distance. Euclidean distance* adalah perhitungan jarak dari 2 buah titik dalam *Euclidean space. Euclidean space* diperkenalkan oleh seorang matematikawan dari Yunani sekitar tahun 300 B.C.E. untuk mempelajari hubungan antara sudut dan jarak. *Euclidean distance* ini biasanya diterapkan pada 2 dimensi dan 3 dimensi. Walaupun demikian, metode ini juga dapat diterapkan untuk dimensi yang lebih tinggi. Rumusan *euclidean distance* adalah [LAR-05] :

$$d_{ij} = \sqrt{\sum_{k=1}^{n}(x_{ik} - x_{jk})^2} \quad (2\text{-}13)$$

dengan :

d_{ij} = jarak data i dengan data j.

k = dimensi ke- k.

n = jumlah dimensi.

x_{ik} = data i dimensi ke k.

x_{jk} = data j dimensi ke k.

2.2.12 Metode Gabungan *K-Means* dan LVQ

Metode gabungan *K-Means* dan LVQ merupakan metode yang menggabungkan *K-Means* yang merupakan metode clustering dengan LVQ yang merupakan metode klasifikasi. Metode ini dikembangkan oleh Dian Eka Ratnawati, dkk, yang dilakukan di Universitas Brawijaya pada tahun 2012 dengan penelitian yang berjudul "Pengembangan Metode Klasifikasi berdasarkan *K-Means* dan LVQ". Algoritma metode ini adalah [RAT-12] :

1. Data semula disajikan dalam bentuk tabel yang disimpan dalam database.
2. Mengelompokkan data dengan metode *K-Means*.
3. Dilakukan pengecekan terhadap hasil *clustering* tersebut, jika data pada *cluster* tersebut mempunyai kelas yang berbeda maka dilihat jumlah *record.*
4. Jika jumlah *record* > *treshold*, maka ulangi langkah 2.
5. Jika jumlah *record* <= *treshold*, dilakukan dengan metode LVQ, maka akan didapatkan *centroid* tiap kelas.
6. Bobot hasil pelatihan yang digunakan adalah pusat *cluster* untuk *cluster* yang didapat dari metode *K-Means* atau bobot hasil metode LVQ *cluster* yang didapat dari metode LVQ.
7. Data uji diukur jaraknya dengan menggunakan *euclidean distance* berdasarkan bobot yang didapat dari pelatihan. Data uji dimasukkan dalam kelas yang sama dengan data latih dengan jarak terdekat

2.2.7 Akurasi Sistem

Akurasi merupakan ukuran seberapa dekat suatu angka hasil pengukuran terhadap angka sebenarnya. Akurasi dapat diperoleh dari persentase kebenaran, yaitu perbandingan antara jumlah data benar dengan keseluruhan data [LUD-11]. Akurasi dinyatakan dengan persamaan :

$$akurasi = \frac{jumlah\ data\ benar}{jumlah\ data} * 100\% \quad (2\text{-}14)$$