BAB 3 PERANCANGAN

Bab ini menjelaskan tentang perancangan yang digunakan untuk melakukan eksperimen klasifikasi dokumen teks. Bab perancangan klasifikasi dokumen teks ini meliputi data (subbab 3.1), persiapan dokumen teks (subbab 3.2), pembobotan kata (subbab 3.3), pengembangan ontologi (3.4) dan metode klasifikasi dokumen teks (subbab 3.5) yaitu Naïve Bayes (subbab 3.5.1) dan ontologi (subbab 3.5.2).

3.1 Data

Data (dokumen teks) yang digunakan dalam penelitian ini berjumlah 1300 dokumen berita berbahasa Indonesia yang diperoleh dari situs http://www.kompas.com. Dokumen uji coba tersebut merupakan kumpulan berita dari kategori olahraga. Jenis dokumen uji coba terdiri dari dokumen bulutangkis, basket, sepakbola, tenis, dan otomotif. Rincian daftar kategori dan jumlah dokumen teks yang digunakan dalam penelitian ini dapat dilihat pada Tabel 3.1. Contoh dokumen teks yang digunakan pada penelitian ini dapat dilihat pada Lampiran 2.

Tabel 3.1 Daftar Kategori dan Jumlah Dokumen Teks

Kategori	Jumlah Dokumen
Bulutangkis	155
Basket	59
Otomotif	160
Sepakbola	767
Tenis	159

3.2 Persiapan Dokumen Teks

Tahap ini merupakan tahap yang dilakukan sebelum menjalankan metode klasifikasi dokumen teks. Tahap ini bertujuan untuk menyeragamkan bentuk kata,

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

menghilangkan karakter-karakter selain huruf, dan mengurangi volume kosakata. Contoh dokumen masukan dapat dilihat pada Gambar 3.1 dan Gambar 3.2.

Budi berbelanja di warung. Budi membeli roti, selai, dan susu.

Gambar 3.1 Contoh Dokumen Masukan 1

Badu dan budi pergi dengan bus. Ayah badu seorang supir bus.

Gambar 3.2 Contoh Dokumen Masukan 2

Tahap persiapan dokumen teks terdiri dari empat tahap, yaitu *case folding* (subbab 3.2.1), tokenisasi (subbab 3.2.2), pembuangan *stopwords* (subbab 3.2.3), dan pemotongan imbuhan (subbab 3.2.4). Berikut ini merupakan contoh proses persiapan dokumen teks.

3.2.1 *Case Folding*

Tahap *case folding* dilakukan pengubahan huruf dalam dokumen teks menjadi huruf kecil ('a' sampai dengan 'z'). Karakter lain selain huruf dianggap sebagai *delimiter* sehingga karakter tersebut akan dihapus dari dokumen teks. Contoh hasil dokumen teks setelah dilakukan *case folding* dapat dilihat pada Gambar 3.3 dan Gambar 3.4.

budi berbelanja di warung
budi membeli roti selai dan susu

Gambar 3.3 Tahap *Case Folding* Dokumen 1

Badu dan budi pergi dengan bus
ayah badu seorang supir bus

Gambar 3.4 Tahap *Case Folding* Dokumen 2

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

3.2.2 Tokenisasi

Tahap tokenisasi dilakukan pemecahan kalimat menjadi kata-kata. Contoh hasil dokumen teks setelah dilakukan tokenisasi dapat dilihat pada Gambar 3.5 dan Gambar 3.6.

budi
berbelanja
di
warung
budi
membeli
roti
selai
dan
susu

Gambar 3.5 Tahap Tokenisasi Dokumen 1

badu
dan
budi
pergi
dengan
bus
ayah
badu
seorang

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

supir
bus

Gambar 3.6 Tahap Tokenisasi Dokumen 2

3.2.3 Pembuangan *Stopwords*

Tahap ini dilakukan pembuangan kata-kata yang terdapat pada basis data *stopwords*. Contoh hasil dokumen teks setelah dilakukan pembuangan *stopwords* dapat dilihat pada Gambar 3.7 dan Gambar 3.8.

budi
berbelanja
warung
budi
membeli
roti
selai
susu

Gambar 3.7 Tahap Pembuangan *Stopwords* Dokumen 1

badu
budi
pergi
bus
ayah
badu
seorang
supir
bus

Gambar 3.8 Tahap Pembuangan *Stopwords* Dokumen 2

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

3.2.4 Pemotongan Imbuhan

Tahap ini dilakukan pemotongan imbuhan pada kata. Contoh pemotongan imbuhan pada kata “membeli” menjadi kata “beli”. Contoh hasil dokumen teks setelah dilakukan pemotongan imbuhan dapat dilihat pada Gambar 3.9 dan Gambar 3.10.

Gambar 3.9 Tahap Pembuangan Imbuhan Dokumen 1

Gambar 3.10 Tahap Pembuangan Imbuhan Dokumen 2

3.3 Pembobotan Kata

Tahap ini dilakukan setelah tahap pembuangan *stopwords* dan pemotongan imbuhan kata pada setiap dokumen teks yang ada di koleksi dokumen. Pembobotan kata pada

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

tahap ini menggunakan metode TF-IDF. Ada tiga langkah yang perlu dilakukan untuk memperoleh nilai TF-IDF untuk masing-masing kata pada dokumen teks yaitu menghitung nilai TF, IDF, dan TF-IDF. Penghitungan nilai TF pada masing-masing dokumen teks direpresentasikan dalam bentuk *term documents matrix* (Dok_j x TF_{jk}) seperti pada Tabel 3.2.

Tabel 3.2 *Term Documents Matrix* (Dok_j x TF_{jk})

Dok_i	TF1	TF2	...	TFm
Dok1	2	4	...	5
Dok2	2	3	...	2
...	...	...	...	...
Dokn	1	3	...	7

Dok_j merepresentasikan setiap dokumen teks yang ada di koleksi dokumen dimana j=1,...,n. Frekuensi kata TF_{jk} adalah jumlah kemunculan kata w_k pada dokumen Dok_j dimana k=1,...,m. Penghitungan bobot kata TF-IDF_{jk} (x_{jk}) untuk setiap kata w_k dilakukan dengan menggunakan Persamaan (3.1) [SAL83].

$$x_{jk} = TF_{jk} \times IDF_k \quad \textbf{(3.1)}$$

dimana frekuensi dokumen DF_k adalah jumlah semua dokumen teks yang mengandung kata w_k. Penghitungan invers frekuensi dokumen IDF_k dilakukan dengan menggunakan frekuensi dokumen DF_k seperti yang terlihat pada Persamaan (3.2).

$$IDF_k = \log\left(\frac{n}{DF_k}\right) \quad \textbf{(3.2)}$$

dimana n adalah jumlah semua dokumen teks yang ada dikoleksi dokumen.

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

Berdasarkan *term documents matrix* seperti pada Tabel 3.2, kita dapat membuat *term documents matrix* untuk dokumen 1 (Gambar 3.1) dan dokumen 2 (Gambar 3.2) setelah melakukan tahap persiapan dokumen seperti pada Tabel 3.3.

Tabel 3.3 *Term Documents Matrix* untuk Dokumen 1 dan Dokumen 2

	Dokumen$_1$	**Dokumen$_2$**
budi	2	1
belanja	1	0
warung	1	0
beli	1	0
roti	1	0
selai	1	0
susu	1	0
badu	0	2
pergi	0	1
bus	0	2
ayah	0	1
orang	0	1
supir	0	1

Setelah membuat *term documents matrix* untuk dokumen 1 dan dokumen 2, kita menghitung nilai frekuensi dokumen DF_k dan invers frekuensi dokumen IDF_k untuk setiap w_k pada dokumen 1 dan dokumen 2 dimana k=1,...,13 dengan menggunakan persamaan (3.2) dan n (jumlah dokumen) = 2 seperti pada Tabel 3.4.

Tabel 3.4 DF_k dan IDF_k untuk Dokumen 1 dan Dokumen 2

k	**Kata**	**DF_k**	**IDF_k**
1	budi	2	log(2/2)
2	belanja	1	log(1/2)
3	warung	1	log(1/2)

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

4	beli	1	log(1/2)
5	roti	1	log(1/2)
6	selai	1	log(1/2)
7	susu	1	log(1/2)
8	badu	1	log(1/2)
9	pergi	1	log(1/2)
10	bus	1	log(1/2)
11	ayah	1	log(1/2)
12	orang	1	log(1/2)
13	supir	1	log(1/2)

Setelah menghitung nilai frekuensi dokumen DF_k dan invers frekuensi dokumen IDF_k, kita menghitung nilai TF-IDF_{jk} (x_{jk}) untuk setiap w_k pada $dokumen_j$ dimana j=1,2 dan k=1,...,7 dengan menggunakan Persamaan (3.1) seperti pada Tabel 3.5 dan Tabel 3.6.

Tabel 3.5 Nilai TF-IDF_{1k}

k	kata	TF-IDF_{1k}
1	budi	2 x log(2/2)
2	belanja	1 x log(1/2)
3	warung	1 x log(1/2)
4	beli	1 x log(1/2)
5	roti	1 x log(1/2)
6	selai	1 x log(1/2)
7	susu	1 x log(1/2)

Tabel 3.6 Nilai TF-IDF_{2k}

k	kata	TF-IDF_{1k}
1	budi	1 x log(2/2)
2	badu	2 x log(1/2)

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

3	pergi	1 x log(1/2)
4	bus	2 x log(1/2)
5	ayah	1 x log(1/2)
6	orang	1 x log(1/2)
7	supir	1 x log(1/2)

3.4 Pemodelan Ontologi

Ontologi dapat bermakna informasi yang digunakan dalam domain khusus dan mendefinisikan hubungan dalam bentuk relasi antara satu informasi dengan informasi lainnya. Dalam penelitian ini, ontologi dikembangkan dengan menerapkan tahapan pengembangan ontologi seperti yang terdapat pada subbab 2.4. Tahapan yang dilakukan dalam proses pengembangan ontologi adalah sebagai berikut.

1. Tahap penentuan domain

   Domain yang dipilih untuk mengembangkan ontologi dalam penelitian ini adalah domain olahraga. Domain olahraga untuk pengembangan ontologi disesuaikan dengan data yang ada pada subbab 3.1.

2. Tahap penggunaan ulang ontologi

   Ontologi dalam penelitian ini dikembangkan dengan memanfaatkan ontologi yang sudah ada di IPTC NewsCodes. NewsCodes merupakan kumpulan konsep berbentuk hierarki yang dapat digunakan untuk melakukan klasifikasi artikel berita. NewCodes terdiri dari 1400 konsep, yang disusun dalam tiga tingkatan *(Subject, SubjectMatter, SubjectDetail)*. Konsep tersebut dilakukan perbaikan dan perluasan agar dapat digunakan untuk klasifikasi dokumen teks.

3. Tahap penyebutan istilah-istilah pada ontologi

   Beberapa istilah penting yang muncul pada dokumen teks olahraga seperti kejuaraan, tim, pemain, pelatih dan istilah-istilah yang digunakan dalam permainan olahraga. Istilah penting ini yang kemudian didefinisikan sebagai *class* pada ontologi.

4. Tahap pendefinisian *class* dan hierarki *class*

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

Berdasarkan data yang ada pada subbab 3.1, lima kategori yang ada digunakan untuk mendefinisikan *class.* Kelima kategori tersebut adalah bulutangkis, basket, otomotif, sepakbola, dan tenis. Kelima kategori tersebut merupakan *subclass* dari *class* olahraga dan terdiri dari beberapa *subclass*. Contoh *class* "bulutangkis" memiliki *subclass* "kejuaraan", "pemain", dan "istilah".

5. Tahap pendefinisian *property*

   Pada tahap ini kita mendefinisikan *property* untuk masing-masing kelas yang ada di ontologi. Misalkan *class* "pemain" terdiri dari beberapa *property* seperti "nama_pemain" dan "kewarganegaraan_pemain". Selain itu, kita juga dapat mendefinisikan *property* antar *class* seperti "mengikuti_kejuaraan", *property* ini menghubungkan *class* "pemain" dengan *class* "kejuaraan".

6. Tahap pendefinisian *facets*

   Pada tahap ini kita mendefinisikan *facets* dari setiap *property* yang ada di *class* ontologi. Misalkan *class* "pemain" yang terdiri dari *property* "nama_pemain", "kewarganegaraan_pemain", dan "mengikuti_kejuaraan". *Facets* untuk *property* "nama_pemain" dan "kewarganegaraan_pemain" adalah *string*, sedangkan *facets* untuk *property* "mengikuti_kejuaraan" adalah *minimun cardinality* = 1 dan *multiple cardinality* = N (satu pemain dapat mengikuti banyak kejuaraan).

7. Tahap pendefinisan *instances*

   Pada tahap ini, kita membuat *instance* dari setiap *class* yang mungkin di ontologi. Contoh *instance* untuk *class* "pemain" adalah sebagai berikut.

   - nama_pemain: taufik hidayat
   - kewarganegaraan_pemain: indonesia
   - mengikuti_kejuaraan: all england

3.5 Metode Klasifikasi Dokumen Teks

Penelitian ini menggunakan dua metode untuk melakukan klasifikasi dokumen teks, yaitu metode *machine learning* (metode Naïve Bayes) dan ontologi dijelaskan lebih

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

lanjut pada subbab 3.4.1 dan 3.4.2. *Term document matrix* yang telah dihasilkan sebelumnya akan menjadi masukan untuk kedua pendekatan ini sehingga menghasilkan model probabilistik yang akan digunakan untuk melakukan klasifikasi dokumen teks.

3.5.1 Naïve Bayes

Naïve Bayes merupakan metode *fully supervised learning* yang memerlukan dokumen pembelajaran untuk membangun model probabilistik. Model probabilistik yang dihasilkan akan digunakan untuk menghitung *prior probability* dan *conditional probability* dokumen pengujian untuk menentukan kategori dari dokumen pengujian tersebut. Perancangan klasifikasi dokumen dengan menggunakan metode Naïve Bayes dapat dilihat pada Gambar 3.11. Berikut ini penjelasan proses-proses yang terdapat pada Gambar 3.11.

1. Proses persiapan dokumen teks

   Proses persiapan dokumen teks meliputi *case folding*, tokenisasi, pembuangan *stopwords*, dan pemotongan imbuhan kata menjadi kata dasar (subbab 3.2).

2. *K-fold cross validation*

   K-fold cross validation membagi kumpulan dokumen teks menjadi *k* bagian. Penggunaan *K-fold cross validation* dalam penelitian ini bertujuan untuk menghilangkan bias pada data. Dalam satu *set* eksperimen akan dilakukan *k* buah eksperimen klasifikasi dokumen teks dengan setiap eksperimen menggunakan *k-1* bagian data sebagai data pembelajaran dan satu bagian data sebagai data pengujian secara bergantian. Kumpulan dokumen teks yang ada diacak urutannya sebelum dimasukkan ke dalam sebuah *fold*. Hal tersebut bertujuan untuk menghindari pengelompokkan dokumen-dokumen yang berasal dari satu kategori tertentu pada sebuah *fold*. Dalam tiap eksperimen, data pengujian yang digunakan akan berbeda, yang didapat dengan menukar data pengujian dengan salah satu bagian data dari data pembelajaran yang belum pernah digunakan sebagai data pengujian. Penghitungan nilai akurasi

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

akhir didapat dengan merata-ratakan nilai akurasi untuk *k* percobaan yang dilakukan.

K-fold cross validation dalam penelitian ini dibuat secara manual. Proses klasifikasi menggunakan 5-*fold cross validation*. Banyaknya data pengujian pada satu kategori dalam sebuah *fold* berjumlah 1/5 dari jumlah total dokumen pada kategori tersebut. Data pembelajaran berjumlah 4/5 dari total dokumen pada setiap kategori. Dengan menggunakan 5-*fold cross validation,* maka untuk masing-masing metode klasifikasi akan dibuat lima data pengujian dan data pembelajaran, dengan variasi sebagai berikut:

- Data pengujian n yang digunakan adalah fold (n-1 modulo 5)+1.
- Data pembelajaran n yang digunakan adalah gabungan fold (n modulo 5)+1, fold (n+1 modulo 5)+1, fold (n+2 modulo 5)+1, dan fold (n+3 modulo 5)+1.

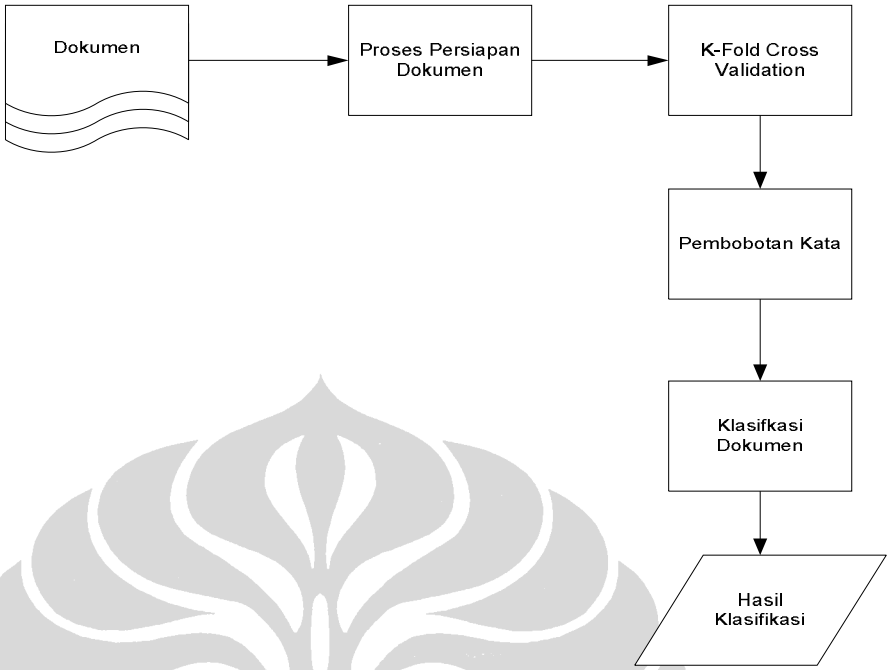
3. Pembobotan kata

   Proses pembobotan kata meliputi proses pembuatan *term documents matrix* serta menghitung nilai TF-IDF (subbab 3.3).

4. Klasifikasi Dokumen Teks

   Proses ini dilakukan setelah melakukan proses persiapan dokumen teks, *k-fold cross validation* dan pembobotan kata. Langkah pertama yang dilakukan sebelum melakukan klasifikasi dokumen teks adalah menentukan kategori $c \varepsilon C=\{$bulutangkis, basket, otomotif, sepakbola, tenis$\}$ dari suatu dokumen $d \varepsilon D=\{d_1, d_2, d_3, \ldots, d_j\}$ berdasarkan kata-kata yang ada di dokumen teks. Kumpulan dokumen pembelajaran dan dokumen pengujian direpresentasikan dalam bentuk *term documents matrix* seperti pada subbab 3.3. Proses menentukan kategori dari sebuah dokumen pengujian dilakukan dengan menggunakan Persamaan (2.7).

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009



Gambar 3.11 Perancangan Klasifikasi Dokumen Teks (Naïve Bayes)

3.5.2 Ontologi

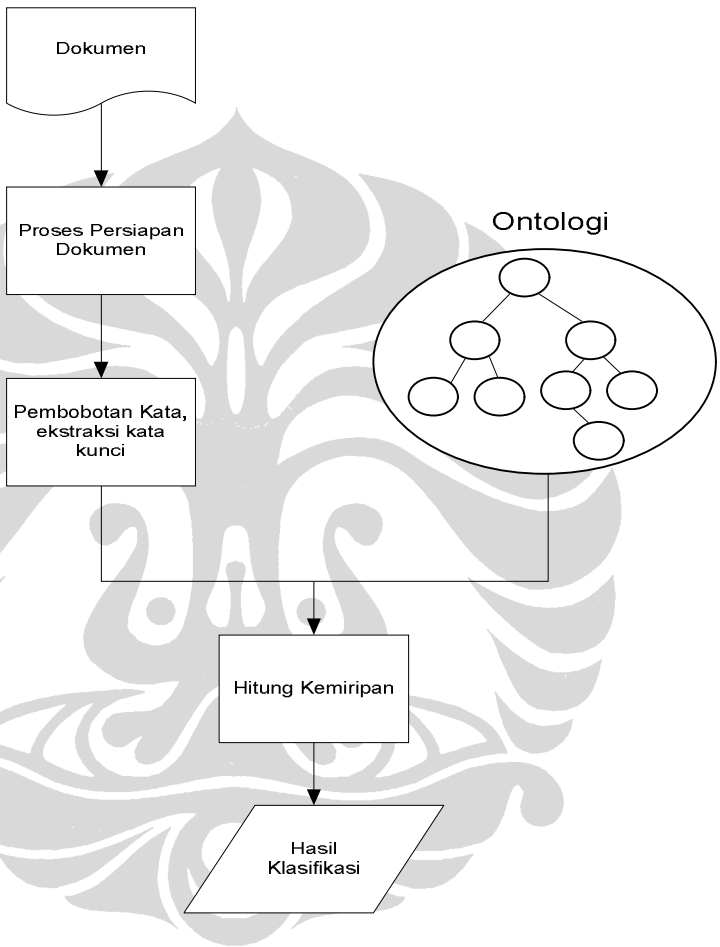
Setiap kategori yang ada di Subbab 3.1 direpresentasikan sebagai ontologi. Proses klasifikasi dokumen teks menggunakan ontologi tidak membutuhkan dokumen pembelajaran. Proses klasifikasi dilakukan dengan memetakan dokumen teks ke sebuah node dengan nilai kemiripan paling tinggi dan dokumen teks tersebut diklasifikasikan tepat ke satu *class*. Rumus untuk menghitung nilai kemiripan dapat dilihat pada Persamaan (3.3).

$$Sim(node, d) = \frac{\sum_{i=0}^{n} freq_{i,d} / \max_{i,d}}{N} \times \frac{V_d}{V} \quad \textbf{(3.3)}$$

dimana N adalah frekuensi fitur dari sebuah *node*, $freq_{i,d}$ merepresentasikan frekuensi fitur dari fitur i yang cocok di dokumen d, $max_{i,d}$ merepresentasikan frekuensi fitur yang paling cocok di dokumen d, V adalah jumlah *constraint*, dan V_d adalah jumlah *constraint* yang terpenuhi di dokumen d. Proses klasifikasi dokumen hanya dilakukan ketika menggunakan relasi "*is-a*" dan "*has-a*". Ketika *node* lain cocok dengan fitur di

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009

dokumen, maka *node* tersebut juga dimasukkan ke dalam proses klasifikasi dokumen untuk menghitung nilai kemiripannya. Proses klasifikasi dokumen dengan menggunakan pendekatan ini, menghasilkan proses klasifikasi dokumen yang lebih akurat. Perancangan klasifikasi dokumen teks dengan menggunakan ontologi dapat dilihatpada Gambar 3.12.



Gambar 3.12 Perancangan Klasifikasi Dokumen Teks (Ontologi)

Pengklasifikasian artikel..., Prajna Wira Basnur, FASILKOM UI, 2009