semanTIK, Vol.3, No.2, Jul-Des 2017, pp. 187-194
ISSN : 2502-8928 (Online)

IMPLEMENTASI *TEXT MINING* KLASIFIKASI SKRIPSI MENGGUNAKAN METODE *NAÏVE BAYES CLASSIFIER*

Apriliana*[1], Natalis Ransi[2], Jumadil Nangi[3]
*[1,2,3]Jurusan Teknik Informatika, Fakultas Teknik, Universitas Halu Oleo, Kendari
e-mail : ***[1]apriliyana43@gmail.com**,[2]natalis.ransi@gmail.com, [3]jumadilnangi87@gmail.com

Abstrak

Pada era perkembangan teknologi saat ini, skirpsi dapat dilihat menggunakan internet seperti TAKP Teknik Informatika yang merupakan salah satu *website* skripsi Teknik Informatika Universitas Halu Oleo yang sering dikunjungi jika telah menyelesaikan skripsi. Kategori skripsi yang digunakan adalah tiga kategori, yaitu Rekayasa Perangkat Lunak, Komputasi berbasis jaringan, dan Komputasi Cerdas Visual, dan dimana data skripsi tersebut diambil dari situs TAKP Teknik Informatika dan Perpustakaan Prodi Teknik Informatika.Terkadang pengklasifikasian kategori skripsi masih menjadi kendala.Untuk mempermudah dalam pengklasifikasian kategori skripsi, diperlukan sebuah sistem dengan menggunakan metode *text mining* sebagai salah satu alternatif untuk menyelesaikannya.

Berdasarkan hasil pengujian, Algoritma *Naïve Bayes Classifier* memiliki kinerja yang baik untuk klasifikasi skripsi. Hal ini dibuktikan pada pengujian manual dan pengujian sistem menggunakan abstrak skripsi kemudian skripsi diklasifikasikan pada 3 kategori yaitu rekayasa perangkat lunak, komputasi berbasis jaringan , dan komputasi cerdas visual. Hasil klasifikasi menggunakan 51 skripsi uji didapatkan akurasi 94,11%.

Kata kunci—Skripsi, *Text Mining*, *Naïve Bayes Classifier*.

Abstract

In the era of technological today, the thesis can be viewed using the internet as websiteTAKP Informatics engineering which is one of the frequently visited website Informatics engineering. thesis categories used are three categories. that is Software Engineering, Network-Based Computing, and Visual Intelligent Computing, And the data of the thesis are taken from TAKP Informatics Engineering and Library of Informatics Engineering, Sometimes classification of thesis categories are still a constraint. To simplify the classification of thesis category, required a system using the method of text mining as an alternative to solve it.

Based on test results, Naïve Bayes classifier algorithm has a good performance for the classification of types of thesis. This is evidenced in manual testing and system testing using thesis data and then classified into three categories of categories of software engineering, network-based computing, and visual intelligent computing. The results of classification using 51 test thesis obtained 94,11% accuracy.

Keywords—*Thesis, Text Mining, Naïve Bayes Classifier.*

1. PENDAHULUAN

Skripsi adalah istilah yang digunakan di pendidikan dasar untuk mengilustrasikan suatu karya tulis ilmiah berupa paparan tulisan hasil penelitian sarjana S1 yang membahas suatu permasalahan atau fenomena dalam bidang ilmu tertentu dengan menggunakan kaidah-kaidah yang berlaku. Berdasarkan kamus besar Bahasa Indonesia, yang dimaksud dengan skripsi adalah karangan ilmiah yang wajib ditulis oleh mahasiswa sebagai bagian dari persyaratan akhir pendidikan akademisnya.

Pada era perkembangan teknologi saat ini, skirpsi dapat dilihat menggunakan internet seperti TAKP Teknik Informatika yang merupakan salah satu *website*skripsi Teknik Informatika Universitas Halu Oleo yang sering dikunjungi oleh mahasiswa jika telah menyelesaikan skripsi. Kategori skripsi yang digunakan adalah tiga kategori, yaitu Rekayasa Perangkat Lunak, Komputasi berbasis jaringan, danKomputasi Cerdas Visual, dan dimana data skripsi tersebut diambil dari situs TAKP Teknik Informatika.Terkadang pengklasifikasian kategori skripsi masih menjadi kendala.

Untuk mempermudah dalam pengklasifikasian kategori skripsi, diperlukan sebuah sistem dengan menggunakan metode *text mining* sebagai salah satu alternatif untuk menyelesaikannya.*Text mining* adalah salah satu teknik yang dapat digunakan untuk melakukan klasifikasi, dimana *text mining* merupakan variasi dari *data mining* yang berusaha menemukan pola yang menarik dari sekumpulan data tekstual yang berjumlah besar. Selain klasifikasi, *text mining* juga digunakan untuk menangani masalah *clustering*, *information extraction*, dan *information retrival* [1].

2. METODE PENELITIAN

2.1 Skripsi

Skripsi adalah karya ilmiah yang ditulis mahasiswa program S1 yang membahas topik atau bidang tertentu berdasarkan hasil kajian pustaka yang ditulis oleh para ahli, hasil penelitian lapangan, atau hasil pengembangan (eksperimen). Dalam pengerjaan skripsi, mahasiswa dibimbing oleh minimal dua orang dosen pembimbing yang ditunjuk oleh perguruan tinggi yang bersangkutan.Pembimbingan ini dimaksudkan agar hasil skripsi mahasiswa berkualitas baik dari segi isi maupun tekniknya penyampaiannya [2].

2.2 *Data Mining*

a. Pengertian Data *Mining*

Data mining adalah serangkaian proses untuk menggali nilai tambah berupa informasi yang selama ini tidak diketahui secara manual dari suatu basis data dengan melakukan penggalian pola-pola dari data dengan tujuan untuk memanipulasi data menjadi informasi yang lebih berharga yang diperoleh dengan cara mengekstraksi dan mengenali pola yang penting atau menarik dari data yang terdapat dalam basis data.

Knowledge Discovery in Database (KDD) merupakan proses pencarian pengetahuan yang bermanfaat dari kumpulan data. Proses KDD bersifat interaktif dan iteratif, meliputi sejumlah langkah dengan melibatkan pengguna dalam membuat keputusan dan dapat dilakukan pengulangan di antara dua buah langkah.*Data mining* merupakan salah satu proses inti yang terdapat dalam *Knowledge Data Discovery* (KDD). Banyak orang memperlakukan *data mining* sebagai sinonim dari KDD, karena sebagian besar pekerjaan dalam KDD difokuskan pada *Data Mining*.Namun, langkah-langkah ini merupakan proses yang penting yang menjamin kesuksesan dari aplikasi KDD.

Secara umum definisi *Data mining* dapat diartikan sebagai berikut :

1. Proses penemuan pola yang menarik dari data yang tersimpan dalam jumlah besar.
2. Ekstraksi dari suatu informasi yang berguna atau menarik (non-trivial, implisit, sebelumnya belum diketahui potensial kegunaannya) pola atau pengetahuan dari data yang disimpan dalam jumlah besar.
3. Ekplorasi dari analisis secara otomatis atau semiotomatis terhadap data-data dalam jumlah besar untuk mencari pola dan aturan yang berarti.

b. Konsep Data *Mining*

Data mining sangat perlu dilakukan terutama dalam mengelola data yang sangat besar untuk memudahkan aktivitas *recording* suatu transaksi dan untuk proses data *warehousing* agar dapat memberikan informasi yang akurat bagi penggunanya.

Alasan utama mengapa *Data mining* sangat menarik perhatian industri informasi dalam beberapa tahun belakangan ini adalah karena tersedianya data dalam jumlah yang besar dan semakin besarnya kebutuhan untuk mengubah data tersebut menjadi informasi dan pengetahuan yang berguna karena sesuai fokus bidang ilmu ini yaitu melakukan kegiatan mengekstraksi atau menambang pengetahuan dari data yang berukuran/ berjumlah besar,

informasi inilah yang nantinya sangat berguna untuk pengembangan. berikut langkah-langkahnya :

1. *Data cleaning* (untuk menghilangkan *noise* data yang tidak konsisten).
2. *Data integration* (di mana sumber data yang terpecah dapat disatukan).
3. *Data selection* (di mana data yang relevan dengan tugas analisis dikembalikan ke dalam *database*).
4. *Data transformation* (di mana data berubah atau bersatu menjadi bentuk yang tepat untuk menambang dengan ringkasan performa atau operasi agresi).
5. *Knowledge Discovery* (proses esensial di mana metode yang intelejen digunakan untuk mengekstrak pola data).
6. *Pattern evolution* (untuk mengidentifikasi pola yang benar-benar menarik yang mewakili pengetahuan berdasarkan atas beberapa tindakan yang menarik) [3].

2.3 *Text Mining*

Text mining merupakan variasi dari *Data mining*yang berusaha menemukan pola yang menarik dari sekumpulan data tekstual yang berjumlah besar.*Text mining* dapat diartikan sebagai penemuan informasi yang baru dan tidak diketahui sebelumnya oleh komputer, secara otomatis mengekstrak informasi dari

Sumber-sumber yang berbeda. Kunci dari proses ini adalah menggabungkan informasi yang berhasil diekstraksi dari berbagai sumber. *text mining* didefinisikan sebagai data yang berupa teks yang biasanya sumber data didapatkan dari dokumen, dengan tujuan adalah mencari kata-kata yang dapat mewakili isi dari dokumen tersebut yang nantinya dapat dilakukan analisa hubungan antar dokumen.

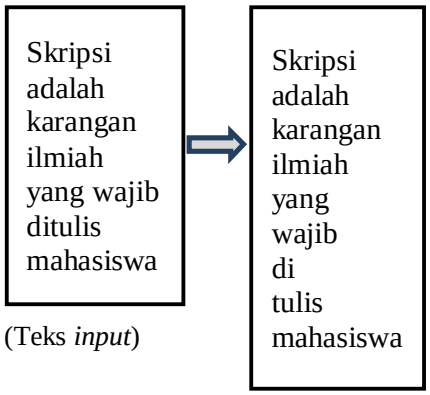
a. Tahapan *Text Mining*

Tahapan *text mining* secara umum dibagi menjadi beberapa tahapan umum

1. *Text Preprocessing*

Text Preprocessing merupakan tahapan awal dari *text mining* yang bertujuan mempersiapkan teks menjadi data yang akan mengalami pengolahan pada tahap selanjutnya. Pada *text mining*, data mentah yang berisi informasi memiliki struktur yang sembarang, sehingga diperlukan proses pengubahan bentuk menjadi data yang terstruktur sesuai kebutuhan, yaitu biasanya akan mejadi nilai-nilai numerik. Proses ini disebut *Text Preprocessing*.

Pada tahap ini, tindakan yang dilakukan adalah *toLowerCase*, dengan mengubah semua karakter huruf menjadi huruf kecil, dan *tokenizing* yaitu proses penguraian deskripsi yang semula berupa kalimat mejadi kata-kata kemudian menghilangkan delimiter-delimiter seperti tanda koma (,), tanda titik (.), spasi, dan karakter angka yang terdapat pada kata tersebut. Gambar 1 menunjukkan Tahap Tokenizing.



(Teks *input*) (Hasil *token*)

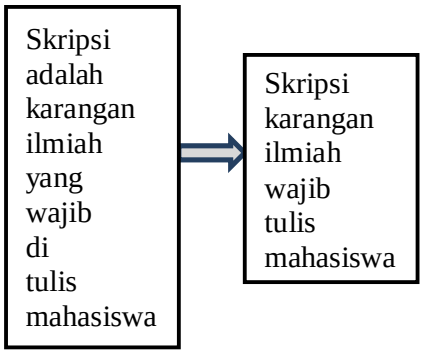
Gambar 1 Contoh Tahap *Tokenizing*

2. Seleksi Fitur (*Feature Selection*)

Pada tahap ini akan dilakukan seleksi dengan mengurangi jumlah kata-kata yang dianggap tidak penting dalam dokumen tersebut untuk menghasilkan proses pengklasifikasian yang lebih efektif dan akurat. Tahapan ini adalah dengan melakukan penghilangan *stopword* dan juga mengubah kata-kata kedalam bentuk dasar terhadap kata yang berimbuhan.

Stopword merupakan kosakata yang bukan merupakan ciri atau kata unik dari suatu dokumen seperti kata sambung. Yang termasuk *stopword* yaitu " di", "pada", "sebuah", "karena", "oleh" dan sebagainya. Sebelum memasuki tahapan penghilang *stopword*, daftar *stopword* harus dibuat terlebih dahulu. Jika kata-kata yang termasuk s*topword* masuk dalam *stoplist*, maka kata tersebut akan dihapus dari deskripsi sehingga sisanya dianggap sebagai kata-kata yang mencirikan isi dokumen atau *keywords*.

Gambar 2 menunjukkan Tahap Seleksi Fitur (*Feature Selection).*



Gambar 2 Contoh Tahap Seleksi Fitur (*Feature Selection*)

3. *Stemming*

Stemming adalah proses pemetaan dan penguraian berbagai bentuk dari suatu kata menjadi kata dasarnya. Tujuan dilakukannya proses *stemming* adalah menghilangkan imbuhan-imbuhan berupa *prefix*, *suffix*, maupun konfiks yang terdapat pada setiap kata. Apabila imbuhan tadi tidak dihilangkan maka setiap kata akan disimpan di dalam *database*, sehingga nantinya akan menjadi beban di dalam *database*. Bahasa Indonesia memiliki aturan morfologi maka proses *stemming* harus berdasarkan aturan morfologi bahasa Indonesia [4].

2.4 Algoritma *Naïve Bayes Classifier*

Algoritma *Naïve Bayes Classifier* merupakan algoritma yang digunakan untuk mencari nilai probabilitas tertinggi untuk mengklasifikasi data uji pada kategori yang paling tepat.Dalam penelitian ini yang menjadi data uji adalah dokumen skripsi.Ada dua tahap pada klasifikasi dokumen.Tahap pertama adalah pelatihan terhadap dokumen yang sudah diketahui kategorinya. Sedangkan tahap kedua adalah proses klasifikasi dokumen yang belum diketahui kategorinya. Persamaan (1) digunakan pada Algoritma *Naïve BayesClassifier*.

$$P\,(A|B) \;=\; (P(B|A) * P(A)) / P(B) \qquad (1)$$

Peluang kejadian *A* dengan syarat *B* ditentukan dari peluang *B* dengan syarat *A*, peluang *A*, dan peluang *B*. Pada pengaplikasiannya nanti, Persamaan (1) dinyatakan menjadi Persamaan (2).

$$P(C_i|D) \;=\; (P(D|C_i) * P(C_i)) \,/\, P(D) \qquad (2)$$

Naïve Bayes Classifier atau bias disebut sebagai multinominal *naïve bayes* merupakan model penyederhanaan dari *TeoremaBayes* yang cocok dalam pengklasifikasian kategori skripsi. Persamaannya dinyatakan dalam Persamaan (3).

$$V_{MAP} \;=\; arg\ max\ P\big(V_j \big| a_1, a_2, \ldots\ldots a_n\big) \qquad (3)$$

Menurut Persamaan (2), maka Persamaan (3) dapat ditulis menjadi Persamaan (4).

$$V_{MAP} \;=\; \underset{V_j \in V}{Arg\ Max} \frac{P\,(a_1, a_2, \ldots a_n | V_j)\, P(V_j)}{P\,(a_1, a_2, \ldots a_n)} \qquad (4)$$

Karena $P(\,a_1,\ a_2, \ldots\ldots a_n)$ konstan, sehingga Persamaan (4) dapat ditulis menjadi Persamaan (5) :

$$V_{MAP} \;=\; \underset{V_j \in V}{Arg\ Max}\, P\,(a1, a2, \ldots an\,|V_j)\, P(V_j) \qquad (5)$$

V_{MAP} = probabilitas kelas *V* atau kelas kategori tertinggi

$P(v_j)$ = peluang jenis kelas *V* atau kategori ke-*j*

$P(a_1, a_2, a_3, \ldots a_n|v_j)$ = peluang atribut jika diketahui keadaan v_j

Namun, karena $P(a_1, a_2, a_3, \ldots a_n|v_j)$ sulit untuk dihitung, maka akan diasumsikan bahwa setiap kata tidak mempunyai keterkaitan, ditunjukkan oleh Persamaan (6).

$$V_{MAP} \;=\; \underset{V_j \in V}{Arg\ Max}\, P(V_j) \prod P\,(a_i|V_j) \qquad (6)$$

Sehingga, perhitungan *Naïve bayes classifier* adalah:

Menghitung $P(ai|vj)$ dengan rumus :

$$P(a_i|v_j) \;=\; \frac{nc_i + m.p}{n + m} \qquad (7)$$

Dengan :

nc_i = kelas kata ke-i yang bernilai ya atau tidak (1 atau 0)

P = 1/ banyaknya kelas *v*

P = Peluang

m = jumlah parameter / total kata

n = jumlah *record* kata pada setiap kelas kategori

Persamaan (7) diselesaikan melalui perhitungan sebagai berikut :

1. Menentukan nilai *nc* untuk setiap *class*
2. Menghitung nilai *P* $(a_i|v_j)$ dan menghitung nilai $P(v_j)$ menggunakan Persamaan (8).

$$V_{MAP} = \underset{V_j \in V}{Arg\ Max}\ P(V_j) \prod P\ (a_i|V_j) \qquad (8)$$

Dengan :$P(a_i|v_j) = \frac{nc+m.p}{n+m}$

3. Menghitung $P(a_i|v_j)\ x\ P(v_j)$ untuk tiap kelas *v*
4. Menentukan hasil klasifikasi yaitu kelas *v* yang memiliki hasil perkalian yang terbesar. [5]

2.5 *Unified Modeling Language*(UML)

Unified Modeling Language (UML) merupakan sistem arsitektur yang bekerja dalam OOAD (*Object-Oriented Analysis/Design*) dengan satu bahasa yang konsisten untuk menentukan, visualisasi, mengkontruksi, dan mendokumentasikan *artifact* (sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa *software*, dapat berupa model, deskripsi, atau*software)* yang terdapat dalam sistem *software*. UML merupakan bahasa pemodelan yang paling sukses dari tiga metode OO yang telah ada sebelumnya, yaitu *Booch*, OMT *(Object Modeling Technique)*, dan OOSE *(Object-Oriented Software Engineering)* [6].

3. PERANCANGAN SISTEM

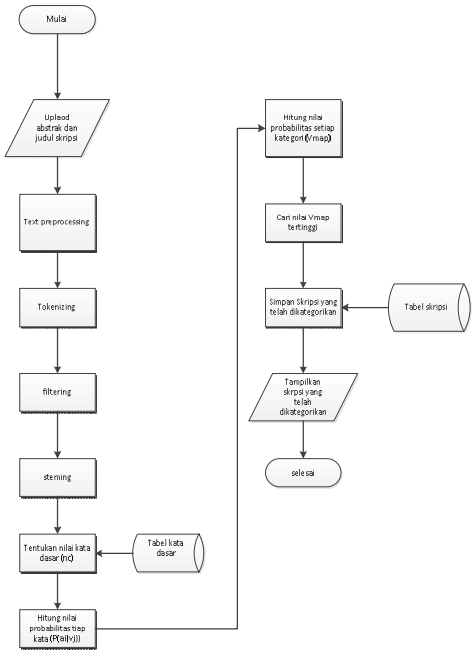
3.1 Analisis Sistem yang Direncanakan

Sistem Implementasi *Text Mining* untuk Klasifikasi Skripsi menggunakan Metode *Naïve Bayes Classifier* merupakan sebuah sistem yang dapat membantu mengklasifikasikankategori skripsi menjadi lebih baik. Sistem ini dapat melakukan proses klasifikasi kategori skripsi berdasarkan abstrak, di mana *user* hanya perlu mengunggah skripsi ke sistem. Kemudian sistem akan melakukan klasifikasi kategori skripsi secara otomatis, agar data tersebut selanjutnyadisimpan di dalam *database*.

1. *Flowchart*

Setelah menganalisis sistem, maka didapatkan *flowchart diagram.* Sistem Implementasi *Text Mining* untuk Klasifikasi Skripsi menggunakan Metode *Naïve Bayes Classifier* yang ditunjukkan oleh Gambar 3. Adapun alur kerja *flowchart diagram* sistem adalah sebagai berikut :

a. *User* memasukkan abstrak dan judul skripsi.
b. Berita melalui proses *text preprocessing*.
c. Berita melalui proses *filtering.*
d. Berita melalui proses *stemming*.
e. Mencari nilai *nc* tiap kata dalam Skripsi, dengan mencocokkan kata pada skripsi dengan kata dalam tabel kata dasar di *database* sistem.
f. Menghitung nilai probabilitas tiap kata ($P(a_i|v_j)$) dalam skripsi.
g. Hitung nilai probabilitas skripsi tiap kategori (V_{MAP}).
h. Membandingkan nilai probabilitas skripsi tiap kategori (V_{MAP}).
i. Menentukan kategori skripsi berdasarkan nilai V_{MAP} tertinggi.
j. Simpan skripsi yang telah ditentukan kategorinya pada tabel skripsi di *database.*



Gambar 3*Flowchart Diagram* Sistem

4. HASIL DAN PEMBAHASAN

4.1 Pengujian Hasil Klasifikasi

Pengujian hasil klasifikasi dilakukan untuk mengetahui tingkat keakurasian sistem

implementasi *text mining* untuk klasifikasi skripsi menggunakan Algoritma *Naïve Bayes Classifier*.

Adapun hasil pengujian klasifikasi sembilan skripsi uji lainnya, ditunjukkan oleh Tabel 1 menunjukkan Pengujian Klasifikasi

Tabel 1 Tabel Pengujian Klasifikasi

Nama File	Label Asli	Jumlah kata dasar dikenali			Presentase	Keterangan
		RPL	KBJ	KCV		
Dokumen 1	RPL	6	0	0	RPL 100% KBJ 0% KCV 0%	Berhasil
Dokumen 2	KCV	5	1	7	RPL 38,46% KBJ7,7 % KCV53, 84%	Gagal
Dokumen 3	RPL	9	0	0	RPL 100% KBJ 0% KCV 0%	Berhasil
Dokumen 4	RPL	16	0	0	RPL 100% KBJ 0% KCV 0%	Berhasil
Dokumen 5	RPL	8	1	0	RPL 89% KBJ 12% KCV 0%	Berhasil
Dokumen 6	KCV	2	0	5	RPL 28,57% KBJ 0% KCV 71,42%	Berhasil
Dokumen 7	RPL	9	0	0	RPL 100 % KBJ 0% KCV 0%	Berhasil
Dokumen 8		1	0	9	RPL 10% KBJ 0% KCV 90%	Berhasil
Dokumen 9	KCV	0	0	2	RPL 0% KBJ 0% KCV 100%	Berhasil
Dokumen 10	RPL				100% KBJ 0% KCV0%	Berhasil
Dokumen 11	RPL	5	0	0	RPL 100 % KBJ 0% KCV 0%	Berhasil
Dokumen 12	KCV	2	0	3	RPL 40% KBJ 0% KCV 60 %	Berhasil
Dokumen 13	RPL	5	0	0	RPL 100% KBJ 0% KCV 0%	Berhasil
Dokumen 14	RPL	5	1	0	RPL 83% KBJ 16,67% KCV 0%	Berhasil
Dokumen 15	RPL	7	0	0	RPL 100% KBJ 0% KCV 0%	Berhasil

4.2 Pengujian *Black Box*

Pengujian *black box* berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program.Tabel 2 menunjukkan Pengujian *Black Box.*

Tabel 2 Pengujian *Black Box*

No	Skenario	Test Case	Harapan	Hasil
1.	Berhasil menganalisis skripsi berdasarkan abstrak	Mengupload salah satu abtrask pada form tambah skripsi	Abstrak yang dipilih setelah terbuka akan menentukan kategori	Berhasil
2.	Mencocokkan hasil perhitungan manual dengan hasil perhitungan sistem menggunakan 9 skripsi	Menghitung Tingkat kecocokan 51 skripsi	Perhitungan manual dan perhitungan yang dilakukan oleh sistem memiliki hasil yang sama	Berhasil

3.	Menguplo ad yang tidak termaksud skripsi teknik informati ka	Menghitungk an kecocokan skripsi	Perhitung an manual dan perhitung an yang dilakukan oleh sistem memiliki hasil yang tidak sama	Tidak berhasil

5. KESIMPULAN

Berdasarkan pembahasan dan evaluasi dari bab sebelumnya, maka kesimpulan yang dapat diambil adalah :

1. Algoritma *Naïve Bayes Classifier* mampu melakukan proses klasifikasi data skripsi secara otomatis dan proses klasifikasi semakinakurat jika data latih yang digunakan dalampembelajaran berjumlah banyak.
2. Algoritma *Naïve Bayes Classifier* memiliki kinerja yang baik untukklasifikasi skripsi berdasarkan abstrak. Hal ini dibuktikan pada pengujian menggunakan data skripsi yang diambil dari perpustakaan prodi teknik informatika, kemudian skripsi diklasifikasikan pada tiga kategori yaitu rekayasa perangkat lunak, komputasi berbasis jaringan, dan komputasi cerdas visual. Hasil klasifikasi menggunakan 51skripsi didapatkan akurasi 90 %.

5. SARAN

Saran untuk pengembangan lebih lanjut terhadap penelitian ini yaitu Diharapkan pada penelitian selanjutnya, sistem ini dapat dikembangkan lagi dengan menggunakan teks berbahasa Inggris sebagai data uji.

DAFTAR PUSTAKA

[1] Harlian, M. 2006. *Machine Learning Text Kategorization.* Austin : University of Texas.

[2] Hearst, M.A. 1999. *Untangling Text Data Mining.Proceeding of ACL.*Maryland.

[3] Huda Miftahul, 2011. *jurnal dialogia Vol.9, No.2.* Universitas Islam Negeri, Surabaya.

[4] Tan, Pang-Ning. 2006. *Introduction to Data Mining.USA : Pearson Addison Wesley..*

[5] Triawati, C. 2009. *Metode Pembobotan Statistical Concept Based untuk Klastering dan Kategorisasi Dokumen Berbahasa Indonesia.*Institut Teknologi Telkom, Bandung.

[6] Kurniawan, B., Effendi, S., Sitompu, O.S. 2012. *Klasifikasi Konten Berita Dengan Metode Text Mining.*Universitas Sumatera Utara, Medan.