JURNAL PINTER
VOL. 1 NO. 1 JUNI 2017
e-ISSN: 2597- 4475
http://doi.org/10.21009/pinter.1.1.5

Klasifikasi Dokumen Karya Akhir Mahasiswa Menggunakan Naïve Bayes Classifier (NBC) Berdasarkan Abstrak Karya Akhir Di Jurusan Teknik Elektro Universitas Negeri Jakarta

Nur Indah Pratiwi, Widodo
Universitas Negeri Jakarta
nurindahpratiwi71@gmail.com, widodo03@yahoo.com

ABSTRAK

Dokumen karya akhir di Jurusan Teknik Elektro Universitas Negeri Jakarta setiap tahunnya bertambah, pengklasifikasian dokumen menjadi hal yang sangat penting untuk mengorganisasikan dokumen sehingga dapat memudahkan pencarian. Pengembangan Sistem klasifikasi dokumen bertujuan untuk mengembangkan sebuah sistem yang dapat mengklasifikasikan dokumen karya akhir mahasiswa berdasarkan abstrak karya akhir menggunakan algoritma *Naïve Bayes Classifier* (NBC*).* Sehingga, dapat memudahkan pengklasifikasian dokumen karya akhir di Jurusan Teknik Elektro. Dalam penelitian ini menggunakan metode eksperimen dan menggunakan 100 dokumen abstrak, 90 dokumen sebagai *data train* dan 10 dokumen sebagai *data test*. Data diambil dari skripsi mahasiswa Jurusan Teknik Elektro Universitas Negeri Jakarta dari 14 Maret 2014 sampai dengan 27 Maret 2014. Setelah melakukan proses pengembangan perangkat lunak, dihasilkan sebuah sistem klasifikasi yang bernama Sistem Klasifikasi Dokumen Skripsi. Sistem di implementasi menggunakan PHP dan MySQL, dan diuji menggunakan K-*Fold Cross Validation* (10 *Fold*). Berdasarkan pada hasil uji Sistem didapatkan hasil tingkat akurasi sebesar 81%. Oleh karena itu, dapat disimpulkan bahwa Sistem Klasifikasi Dokumen Abstrak Karya Akhir Menggunakan Algoritma *Naïve Bayes* di Jurusan Teknik Elektro telah berhasil dikembangkan.

Kata kunci: *sistem, Naïve Bayes Classifier, klasifikasi, dokumen, dan algoritma.*

1. PENDAHULUAN

Dalam proses pencarian, mahasiswa akan dihadapkan oleh beberapa kategori dokumen karya akhir yang akan dipilih. Proses pencarian yang mudah akan membuat mahasiswa secara efisien mencari referensi atas kategori bidang ilmu yang membuatnya tertarik. Namun sebaliknya, jika proses pencarian berdasarkan kategori tidak terstruktur dengan baik, mahasiswa akan memakan waktu yang lama untuk mencari satu referensi. Sehingga, sistem pengklasifikasi karya akhir yang baik sangat dibutuhkan untuk membantu mahasiswa mengembangkan suatu arah penelitian.

Pengklasifikasian dokumen di Jurusan Teknik Elektro Universitas Negeri Jakarta (UNJ) saat ini masih dilakukan secara manual. Ketika mahasiswa mencari suatu informasi berbentuk dokumen dengan membacanya terlebih dahulu, hal tersebut dapat memakan waktu lebih lama jika informasi yang dicari tidak sesuai dengan apa yang ia harapkan. Sebuah sistem klasifikasi dokumen dapat dikembangan untuk mengklasifikasikan abstrak sebuah karya akhir, yang selanjutnya untuk menentukan karya akhir tersebut termasuk dalam kategori apa.

Oleh karena itu, dibutuhkan suatu sistem yang dapat mengklasifikasikan

Avalaiable at:
http://journal.unj.ac.id/unj/index.php/pinter/article/view/1892

e-ISSN: 2597- 4475
http://doi.org/10.21009/pinter.1.1.5

dokumen berdasarkan abstrak karya akhir dengan cepat dan relevan sehingga *user* dapat menemukan informasi yang dicari dalam waktu yang efisien.

2. NATURAL LANGUAGE PROCESSING

Natural Language Processing (NLP) adalah pemrosesan bahasa alami yang merupakan salah satu tujuan jangka panjang *Artificial Intelligence* (Kecerdasan Buatan) yang digunakan untuk membuat suatu program yang dapat memiliki kemampuan memahami bahasa manusia.[1]

Tujuan dalam bidang *Natural Language* ini adalah melakukan proses pembuatan model komputasi dari bahasa, bahasa yang dihasilkan ini dapat dimengerti oleh komputer sehingga dapat terjadi suatu interaksi antara manusia dan komputer dengan perantara bahasa alami.

Ada 3 (tiga) aspek utama pada teori pemahaman mengenai *Natural Language,* yaitu:

1. **Sintaksis**: yaitu pemahaman tentang urutan kata dalam pembentukan kalimat dan hubungan antar kata tersebut dalam proses perubahan bentuk dari kalimat menjadi bentuk yang sistematis. Meliputi proses pengaturan tata letak suatu kata dalam kalimat akan membentuk kalimat yang dapat dikenali.
2. **Semantik**: yaitu pemetaan bentuk struktur sintaksis dengan memanfaatkan tiap kata ke dalam bentuk yang lebih mendasar dan tidak tergantung struktur kalimat. Semantic mempelajari arti suatu kata dan bagaimana sekumpulan arti kata tersebut membentuk suatu arti kata dari kalimat yang utuh. Dalam tingkatan ini belum tercakup konteks dari kalimat tersebut.
3. **Pragmatik**: pengetahuan pada tingkatan ini berkaitan dengan masing-masing konteks yang berbeda tergantung pada situasi dan tujuan pembuatan sistem.[2]

Berikut ini adalah bidang-bidang pengetahuan yang berhubungan dengan *Natural Language*[3]:

1. **Fonetik dan fonologi**: berhubungan dengan suara yang menghasilkan kata yang dapat dikenali. Bidang ini menjadi penting dalam proses aplikasi yang memakai metode *speech based system*.
2. **Morfologi**: yaitu pengetahuan tentang kata dan bentuknya dimanfaatkan untuk membedakan satu kata dengan lainnya. Pada tingkat ini juga dapat dipisahkan antara kata dan elemen lain seperti tanda baca. Contoh: Melarikan (word)
   Lari (root)
   Me- (prefix)
   -kan (suffix)

2.1. *Information Retrieval*

Pengertian dari kata *Information Retrieval* dapat berarti sangat luas. Dalam kasus pembelajaran, *Information Retrieval* dapat diartikan sebagai berikut: "*Information retrieval* adalah menemukan suatu bahan yang biasanya berbentuk dokumen dari suatu data yang tidak terstruktur yang dapat memenuhi kebutuhan informasi dari sebuah penyimpanan yang besar dan biasanya disimpan di dalam komputer."[4]

Information retrieval akan berhubungan dengan bagaimana cara untuk menyimpan, merepresentasikan, serta mengorganisasikan dan mengakses sebuah kebutuhan informasi. Kebanyakan cara yang digunakan untuk melakukan *information retrieval* adalah dengan menggunakan *keyword* yang ingin dicari, lalu dengan *keyword* tersebut, dibandingkan dengan isi dokumen, setelah itu dihasilkanlah dokumen-dokumen yang relevan dan tidak relevan.

Pada sistem *information retrieval,* terdapat banyak model yang dapat digunakan. Dengan banyaknya model-

e-ISSN: 2597- 4475
http://doi.org/10.21009/pinter.1.1.5

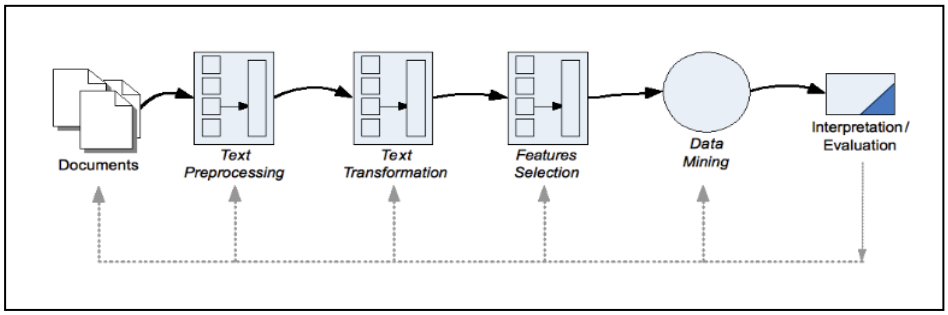
model tersebut untuk melakukan *retrieval* informasi, semakin banyak pula pertimbangan yang dilakukan untuk memilih dokumen yang tepat dengan implementasi *information retrieval* yang diinginkan peneliti. Cara yang paling mudah untuk mendapatkan hasil *retrieval* yang bagus adalah dengan menggabungkan semua fitur yang ada. Namun jika hal tersebut dilakukan, akan memakan waktu yang lebih lama dari proses *indexing* dan *retrieval*-nya itu sendiri.

Secara garis besar, permasalahan yang terjadi pada saat ini dalam masalah data adalah sebagai berikut:

1. Jumlah dokumen digital semakin bertambah dari segi kuantitas.
2. Isi dokumen digital yang semakin banyak, sehingga diperlukan metode paling efektif untuk mengatur dan me-*retriev* kembali data yang telah disimpan.
3. Kesalahan dalam pencarian karena penggunaan metode yang digunakan tidak sesuai.

2.2. *Text Mining*

Text mining atau yang biasa disebut dengan *Text Data Mining* (TDM) merupakan suatu proses pengambilan informasi dari teks yang terdapat di dalamnya[5]. Dengan *text mining*, dapat dicari kata-kata yang dapat mewakili isi dari suatu dokumen, lalu ditentukan kategorinya berdasarkan frekuensi kata-kata yang terdapat di dalamnya. Setelah suatu dokumen dilakukan analisis, maka akan muncul kategori olahraga, kesehatan, selebriti, kriminal, ekonomi, politik atau yang lain, dicocokkan dengan database kata kunci yang sebelumnya telah dibuat.



Gambar 1 Proses Text Mining

3. EKSTRAKSI DOKUMEN

Sebuah dokumen digital mempunyai ketidakstrukturan teks karena dimensinya yang tinggi. Sebelum mengubahnya ke dalam bentuk yang jauh lebih terstruktur diperlukan proses *text mining* terhadap dokumen tersebut. Secara umum proses *text mining* melewai tahap: *Case folding*, *tokenizing*, *filtering*, dan *stemming*.

Case folding adalah mengubah seluruh huruf yang terdapat di dalam dokumen menjadi huruf kecil (*lowercase*). Sedangkan tahap *tokenizing* adalah tahap pemotongan string input berdasarkan tiap kata yang menyusunnya.

Filtering adalah tahap mengambil kata-kata penting dari hasil token. Bisa menggunakan algoritma *stoplist* (membuang kata yang kurang penting) atau *wordlist* (menyimpan kata penting).

Stemming adalah tahap mencari root kata atau kata dasar dari tiap kata hasil filtering. *Stemming* merupakan suatu proses yang terdapat dalam sistem IR yang mentransformasikan kata-kata yang terdapat dalam suatu dokumen ke kata-kata akar atau dasarnya (root word) dengan menggunakan aturan-aturan tertentu.

4. DATA MINING

Data mining adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam *database* dan *data*

e-ISSN: 2597- 4475
http://doi.org/10.21009/pinter.1.1.5

mining juga merupakna proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai *database* besar.

Berdasarkan pada definisi-definisi yang telah disebutkan oleh para ahli di atas, hal penting yang terkait dengan *data mining* adalah[6]:

a. *Data mining* merupakan suatu proses otomatis terhadap data yang sudah ada, tetapi proses tradisional pun masih digunakan.
b. Data yang akan diproses berupa data yang sangat besar.
c. Tujuan *data mining* adalah mendapatkan hubungan atau pola yang mungkin memberikan indikasi yang bermanfaat.

4.1. Klasifikasi Dokumen

Klasifikasi merupakan salah satu tugas penting dalam *data mining*. Pada klasifikasi kelompok data (*class label*) yang sudah diketahui, sebuah data akan masuk ke dalam kelompok tertentu yang sebelumnya telah ditentukan. Setiap hari, jumlah dokumen semakin bertambah. Diantara berbagai bentuk informasi digital, diperkirakan 80% dokumen digital adalah dalam bentuk teks[7]. Tingginya volume dokumen teks ini dikarenakan aktivitas yang terus meningkat dari berbagai sumber berita dan aktivitas penulisan dokumen akademis dari kegiatan riset, konferensi dan pertemuan-pertemuan ilmiah.

Oleh karena itu, klasifikasi dokumen merupakan masalah yang mendasar namun sangat penting karena manfaatnya dapat mengatasi permasalahan yang telah disebutkan sebelumnya. Sebuah dokumen dapat dikelompokkan ke dalam kategori tertentu berdasarkan kata-kata atau kalimat-kalimat yang ada di dalam dokumen tersebut. Kata atau kalimat yang ada di dalam dokumen memiliki makna tertentu dan dapat digunakan untuk menentukan kategori dari dokumen tersebut. Performa pengklasifikasi biasanya diukur dan dinyatakan dengan galat.

Manfaat dari klasifikasi dokumen adalah untuk mengorganisasikan dokumen. Dengan semakin meningkatkan jumlah dokumen yang bertambah setiap harinya, maka akan lebih mudah mencari informasi dari sebuah dokumen yang telah terorganisasi dan telah dikelompokkan menurut kategorinya masing-masing. Contoh aplikasi penggunaan klasifikasi dokumen teks yang banyak digunakan adalah *e-mail spam filtering*. Pada aplikasi *spam filtering* sebuah *e-mail* diklasifikasikan apakah *e-mail* tersebut termasuk *spam* atau tidak dengan memperhatikan kata-kata yang terdapat di dalam *e-mail* tersebut. Aplikasi ini telah digunakan oleh banyak *e-mail provider*.

Dokumen karya akhir mahasiswa pada penlitian ini adalah dokumen yang berasal dari isi abstrak skripsi mahasiswa jenjang S1 Jurusan Teknik Elektro UNJ sebagai bahan penelitian. Sedangkan klasifikasi dokumen karya akhir mahasiswa pada penelitian ini adalah mengklasifikasikan atau mengkategorikan isi abstrak skripsi jenjang S1 Jurusan Teknik Elektro UNJ berbentuk dokumen dengan format .doc atau .docx.

4.2. Algoritma *Naïve Bayes Classifier*

Naïve Bayes merupakan algoritma data mining untuk klasifikasi yang tidak menggunakan *rules* maupun *decision tree.* Karena itulah algoritma *Naïve Bayes* masuk ke dalam kelompok algoritma klasifikasi non-*rule based classification.*[8]

$$V_{nb} = argmax\, v_j \in V\, P\,(v_j) \prod P\,(a_i\,|v_j)$$

Algoritma *Naïve Bayes Classifier* merupakan salah satu algoritma *data mining* untuk klasifikasi, serta merupakan algoritma yang tidak menggunakan *rules* ataupun *decision tree.* Selain itu, NBC memiliki komputasi yang mudah, serta memiliki tingkat akurasi tinggi dan *error rate* yang minimum. Algoritma NBC juga mampu menggenerasikan *token* dengan

e-ISSN: 2597- 4475
http://doi.org/10.21009/pinter.1.1.5

pengenalan karakter sehingga mampu diimplementasikan pada *token* dengan bahasa Indonesia. Sehingga cocok pada kasus dokumen karya akhir yang menggunakan bahasa Indonesia sebagai data penelitian.

4.3. Kehandalan Algoritma Naïve Bayes Classifier

Metoda Naïve Bayes classifier merupakan metoda klasifikasi yang berdasar kepada teorema bayes, sebuah teorema yang terkenal di dalam bidang ilmu probabilitas. Selain itu, metoda ini turut didukung oleh ilmu statistika khususnya dalam penggunaan data petunjuk untuk mendukung keputusan pengklasifikasian. Metoda ini sangat luas dipakai dalam berbagai bidang, khususnya dalam proses klasifikasi dokumen. Klasifikasi ini merupakan salah satu teknik dalam data miningyang merupakan kegiatan penunjang dalam bidang sistem informasi. Seperti halnya metoda-metoda lain, metoda Naïve Bayes classifier ini tidaklah 100% sempurna. Ada banyak kelebihan dan kekurangan dari metoda ini, yang dapat menjadi dasar bahan kajian lebih lanjut untuk mendapatkan atau mengembangkan metoda klasifikasi lain, yang dapat bekerja dengan lebih efektif dan efisien, serta mengurangi jumlah titik kelemahan yang dapat disalahgunakan oleh orang lain.

Metode *Naïve Bayes Classifier* dipilih karena *Naive bayesian filtering* memiliki kelebihan dibandingkan dengan metoda *filtering* yang lain, diantaranya adalah[9]:

1. *Bayesian filter* memiliki komputasi yang mudah.
2. *Bayesian* memeriksa *email* secara keseluruhan yaitu memeriksa token di *database spam* maupun *legitimate*.
3. *Bayesian filtering* termasuk dalam *supervised learning* yaitu secara otomatis akan melakukan proses *learning* dari *email* yang masuk.
4. *Bayesian filtering* cocok diterapkan di level aplikasi *client*/*individual user*.
5. *Bayesian filtering* cocok diterapkan pada *binary class* yaitu klasifikasi ke dalam dua kelas.
6. Metode ini *multilingual* dan internasional. *Bayesian filtering* meng*generate* token dengan pengenalan karakter sehingga mampu diimplementasikan pada *email* dengan bahasa apapun.

5. LANGKAH KERJA SISTEM

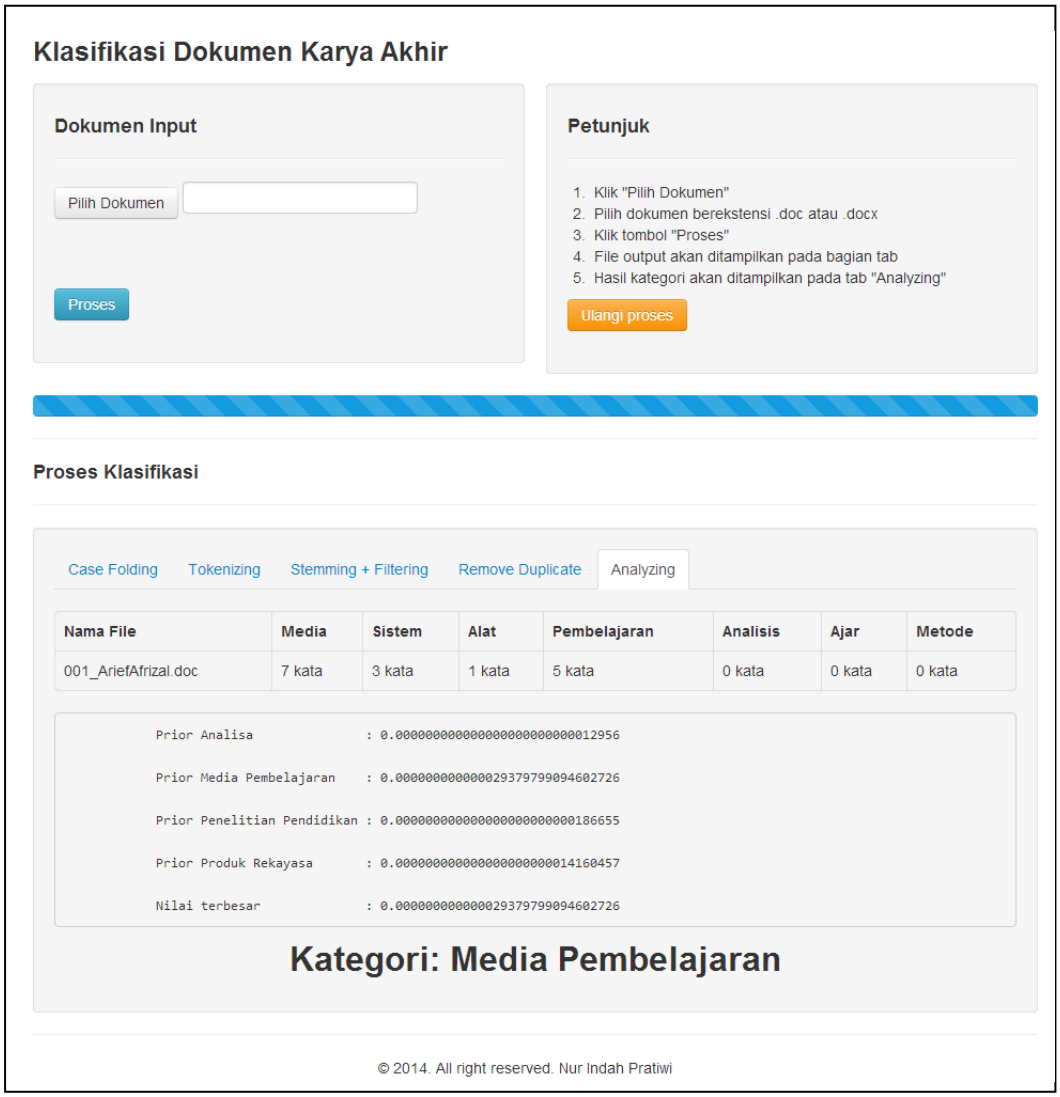
Berikut adalah langkah kerja dalam pengembangan sistem untuk mengklasifikasi dokumen:

1. Mengumpulkan 100 dokumen berbentuk abstrak skripsi
2. Abstrak skripsi tersebut di *filter* sehingga isi abstrak saja yang diambil
3. Pembuatan desain alur data
4. Pembuatan desain *database*
5. Pembuatan desain tampilan sistem
6. Proses *coding* sistem, membuat fungsi *case folding, tokenizing, filtering, stemming* dan *remove duplicate*
7. Input dokumen isi abstrak skripsi ke dalam sistem
8. Menghitung jumlah kata yang muncul
9. Menentukan kata yang akan menjadi acuan
10. Proses perhitungan manual kata tersebut untuk menentukan peluang dalam 10 kali uji
11. Nilai peluang yang dihasilkan akan menjadi variabel pada sistem klasifikasi untuk menghitung peluang pada masing-masing kategori
12. Pembuatan fungsi *analyzing* dengan menggunakan algoritma NBC
13. Nilai peluang yang sudah didapatkan akan dimasukkan ke dalam sistem untuk digabungkan

e-ISSN: 2597- 4475
http://doi.org/10.21009/pinter.1.1.5

dengan fungsi *analyzing* yang telah dibuat

14. Nilai peluang tertinggi akan di tampilkan oleh sistem



Gambar 2 Tampilan halaman result Sistem Pengklasifikasi Dokumen

6. HASIL PENELITIAN

Pengujian sistem menggunakan metode K-*Fold Cross Validation* dengan membagi data menjadi 10 bagian (10 *Fold*) dengan mempertahankan perbandingan pembagian dokumen *data train* dan *data test* sebesar 9:1. Pengujian dan besaran angka akurasi dijelaskan sebagai berikut:

1. Uji ke – 1: *Data Train* berjumlah 90 dokumen dan *Data Test* berjumlah 10 dokumen, dengan tingkat akurasi algoritma *Naïve* Bayes sebesar 70% serta nilai *error* sebesar 30%.
2. Uji ke – 2: *Data Train* berjumlah 90 dokumen dan *Data Test* berjumlah 10 dokumen, dengan tingkat akurasi algoritma *Naïve* Bayes sebesar 90% serta nilai *error* sebesar 10%.
3. Uji ke – 3: *Data Train* berjumlah 90 dokumen dan *Data Test* berjumlah 10 dokumen, dengan tingkat akurasi algoritma *Naïve* Bayes sebesar 70% serta nilai *error* sebesar 30%.
4. Uji ke – 4: *Data Train* berjumlah 90 dokumen dan *Data Test* berjumlah 10 dokumen, dengan tingkat akurasi algoritma *Naïve* Bayes sebesar 80% serta nilai *error* sebesar 20%.
5. Uji ke – 5: *Data Train* berjumlah 90 dokumen dan *Data Test* berjumlah 10

e-ISSN: 2597- 4475
http://doi.org/10.21009/pinter.1.1.5

dokumen, dengan tingkat akurasi algoritma *Naïve* Bayes sebesar 70% serta nilai *error* sebesar 30%.

6. Uji ke – 6: *Data Train* berjumlah 90 dokumen dan *Data Test* berjumlah 10 dokumen, dengan tingkat akurasi algoritma *Naïve* Bayes sebesar 100% serta nilai *error* sebesar 0%.
7. Uji ke – 7: *Data Train* berjumlah 90 dokumen dan *Data Test* berjumlah 10 dokumen, dengan tingkat akurasi algoritma *Naïve* Bayes sebesar 70% serta nilai *error* sebesar 30%.
8. Uji ke – 8: *Data Train* berjumlah 90 dokumen dan *Data Test* berjumlah 10 dokumen, dengan tingkat akurasi algoritma *Naïve* Bayes sebesar 72% serta nilai *error* sebesar 28%.
9. Uji ke – 9: *Data Train* berjumlah 90 dokumen dan *Data Test* berjumlah 10 dokumen, dengan tingkat akurasi algoritma *Naïve* Bayes sebesar 80% serta nilai *error* sebesar 20%.
10. Uji ke – 10: *Data Train* berjumlah 90 dokumen dan *Data Test* berjumlah 10 dokumen, dengan tingkat akurasi algoritma *Naïve* Bayes sebesar 100% serta nilai *error* sebesar 0%.

Data *class* dibagi menjadi empat *class* yaitu kelas pertama dengan nama "Analisis", kelas kedua dengan nama "Media Pembelajaran", kelas kedua dengan nama "Penelitian Pendidikan" dan kelas keempat dengan nama "Produk Rekayasa".

7. KESIMPULAN DAN SARAN

Setelah melalui beberapa tahap pengembangan *software*, mulai dari pengumpulan data abstrak skripsi, proses *filtering* menjadi isi abstrak, hingga proses *preprocessing* serta proses *coding* perangkat lunak, maka dihasilkan sebuah sistem klasifikasi yang bernama Sistem Klasifikasi Dokumen Skripsi.

Berdasarkan hasil penelitian, dihasilkan sebuah *software* yang dapat mengklasifikasikan dokumen berbentuk isi abstrak skripsi untuk mengetahui termasuk ke dalam kategori apa isi abstrak tersebut. Berdasarkan pembahasan di Bab IV, sumber data dokumen abstrak pada penelitian klasifikasi dokumen berasal dari Jurusan Teknik Elektro Universitas Negeri Jakarta. Jumlah data yang diambil berjumlah 100 dokumen, 90 dokumen abstrak digunakan sebagai *data train*, dan 10 dokumen abstrak digunakan sebagai *data test.* Pengujian sistem dilakukan dengan metode *cross validation* dengan hasil sebagai berikut:

Tabel 1 Hasil Pengujian Sistem Klasifikasi Dokumen

Uji Ke -	Akurasi	Rata-Rata Waktu Eksekusi
1	70%	4,2949 sekon
2	90%	4,4857 sekon
3	70%	3,9557 sekon
4	80%	4,2744 sekon
5	80%	4,0553 sekon
6	70%	4,2637 sekon
7	100%	4,2844 sekon
8	70%	4,2628 sekon
9	80%	4,4171 sekon
10	100%	3,7800 sekon
Rata-rata	81%	4,2074 sekon

Untuk penelitian selanjutnya diharapkan dapat mencoba algoritma yang lain, yang kemungkinan mempunyai tingkat akurasi yang lebih tinggi. Sehingga mempunya hasil prediksi yang jauh lebih baik. Serta penambahan jumlah data train akan mempengaruhi tingkat akurasi dari sistem. Maka, diharapkan dalam penelitian selanjutnya target dokumen, baik dokumen set untuk *data training* dan *data test* semakin banyak.

8. DAFTAR PUSTAKA

[1] Raymond J Mooney et al, *UTexas: Natural Language Semantics using Distributional Semantics and*

e-ISSN: 2597- 4475
http://doi.org/10.21009/pinter.1.1.5

Probabilistic Logic, (United States: The University of Texas at Austin, 2014), h.1-2.

[2] D. Poole and Alan Mackworth, *Artificial Intelligence: Natural Language Understanding*, (Canada: Canada License, 2010), h.2.

[3] Derwin Suhartono, *Natural Language Processing*, http://socs.binus.ac.id/2013/06/22/natural-language-processing/, 9 Maret 2014.

[4] Christopher D Manning, *An Introduction to Information Retrieval*, (Cambridge: Cambridge University Press, 2009), h.1.

[5] Ashok Srivastava & Mehran Sahami, Text Mining: Classification, Clustering, and Applications, (Florida: Chapman & Hall CRC Press, 2009), h.12

[6] Kusrini & Emha Taufiq Luthfi, *Algoritma Data Mining*, (Yogyakarta: Penerbit Andi, 2009), h.3

[7] Ah-Hwee Tan, "Text Mining: The state of the art and the challenges", (Singapore, 1999), h.1

[8] Max Brammer, *Principles of Data Mining*, (London: Springer, 2007), h.24.

[9] M. Rachli, "Email Filtering Menggunakan Naive Bayesian", (Bandung: Institut Teknologi Bandung, 2007) h.23