ACADEMIC ARTICLES CLASSIFICATION USING NAIVE BAYES CLASSIFIER (NBC) METHOD

Dwi Pramita B. B[1], Ristu Saptono[2], Rini Anggrainingsih[3]

Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Sebelas Maret
Email: [1]dwi.pbb@student.uns.ac.id, [2]ristu.saptono@staff.uns.ac.id, [3]rini.anggrainingsih@staff.uns.ac.id

ABSTRACT

Sebelas Maret University has been publishing many academic articles. Classifying many articles at a time is not a simple task. The more articles need to be classified, the more energy and time needed. Naive Bayes Classifier method can be used to classify academic articles in short time. Naive Bayes Classifier classifies each article based on the field of study by analyzing its title and abstract. One of feature selection method, Document Frequency Improved (DFM), is implemented for improved the classification performance. This study used of 292 articles as training data and 100 articles as testing data. It tested by applying 5 threshold value from 1 to 2,5 with each threshold executed 5 times. The best results showed at threshold value 2 with the average value of accuracy, precision, recall, and f-measure respectively are 87,8%, 76,6%, 76,2%, and 76,0%.

Keywords: *classification, naive bayes classifier, document frequncy improved*

1. PENDAHULUAN

Publikasi ilmiah atau artikel ilmiah merupakan wadah bagi para dosen dan mahasiswa untuk menyampaikan gagasan ilmiah hasil penelitian dan kajian akademik, dibedakan dalam tiga jenis karya ilmiah yang dapat ditampilkan, yaitu berupa prosiding, jurnal, dan antologi. Jurnal Ilmiah merupakan media yang disediakan oleh lembaga untuk memfasilitasi pemuatan artikel ilmiah dosen. Artikel yang dimuat di jurnal ini dapat digunakan untuk kebutuhan fungsional dosen sebagai tenaga edukatif. Prosiding merupakan kumpulan artikel ilmiah hasil telaah ilmiah yang telah dipresentasikan dalam kegiatan seminar dan sejenisnya baik pada skala regional, nasional, maupun internasional. Antologi adalah kumpulan karya tulis pilihan dari seorang atau beberapa orang pengarang.

Artikel-artikel ilmiah tersebut dapat dikelompokkan berdasarkan bidang ilmu yang sesuai agar tersusun rapi dan memudahkan pencarian. Bidang ilmu terbagi menjadi 15, yaitu *Agriculture & Environment, Biology, Biosciences, Biomedical Research, Clinical and Experimental Medicine I (General & Internal Medicine), Clinical and Experimental Medicine II (Non-Internal Medicine Specialties), Neuroscience & Behavior, Chemistry, Physic, Geosciences & Space Sciences, Engineering, Mathematics, Social Science I (Education, Information, General, Regional & Community Issues), Social Science II (Economics, Business, Management, History, Politics & Law),* dan *Arts & Humanities*[1].

Selama ini di Universitas Sebelas Maret (UNS) pengelompokan atau pengklasifikasian artikel ilmiah dilakukan hanya dengan melihat program studi yang tertera saja. Ketidaksesuaian pengelompokan dengan cara ini bisa saja terjadi jika penulis artikel meneliti masalah di luar cakupan program studinya. Selain itu, mengelompokkan banyak artikel sekaligus tentu bukanlah hal yang mudah dilakukan karena semakin banyak dokumen yang harus dikelompokkan maka tenaga dan waktu yang dibutuhkan juga semakim banyak. Oleh karena itu diperlukan suatu metode yang dapat menemukan informasi mengenai isi artikel dengan cepat. Metode pengolah teks yang diperlukan untuk pengelompokan artikel adalah *text mining*. *Text mining* dapat memberikan informasi dan pengetahuan yang berguna dari sejumlah teks.

Text mining dapat diartikan sebagai penemuan informasi yang baru dan tidak diketahui sebelumnya oleh komputer, dengan secara otomatis mengekstrak informasi dari sumber-sumber yang berbeda. Kunci dari proses ini adalah menggabungkan informasi yang berhasil diekstraksi dari berbagai sumber [2]. Munculnya *text mining* didasarkan pada kenyataan bahwa semakin banyak dokumen yang tersimpan dalam bentuk teks dan kadang dokumen tersebut hanya dibiarkan begitu saja. Padahal jika kumpulan dokumen tersebut diolah lebih lanjut akan didapatkan suatu informasi yang mungkin berguna [3].

Salah satu teknik dalam *text mining* adalah klasifikasi atau kategorisasi teks. Klasifikasi digunakan untuk menempatkan bagian yang tidak diketahui pada data ke dalam kelompok yang sudah diketahui. Klasifikasi menggunakan variabel target dengan nilai nominal. Dalam satu set pelatihan, variabel target sudah diketahui. Dengan pembelajaran dapat ditemukan hubungan antara fitur dengan variabel target.

Metode yang banyak digunakan dalam klasifikasi dokumen salah satunya adalah *Naive Bayes Classifier (NBC)* yang memiliki beberapa kelebihan antara lain, sederhana, cepat dan berakurasi tinggi. Metode *Naive Bayes Classifier (NBC)* untuk klasifikasi atau kategorisasi teks menggunakan atribut kata yang muncul dalam satu dokumen sebagai dasar klasifikasinya. Algoritma klasifikasi *Naive Bayes* memanfaatkan teori probabilitas yang dikemukan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi probabilitas di masa depan berdasarkan pengalaman di masa sebelumya [4].

Kelebihan dari penggunaan Naïve *Bayes Classifier* dalam klasifikasi dokumen dapat ditinjau dari prosesnya yang mengambil aksi berdasarkan data-data yang telah ada sebelumnya. Oleh karena itu, klasifikasi dokumen dengan metode ini dapat disesuaikan sesuai dengan sifat dan kebutuhan masing-masing orang [5].

Seperti yang telah diketahui Universitas Sebelas Maret (UNS) telah banyak menerbitkan artikel ilmiah dari berbagai bidang ilmu. Artikel ilmiah tersebut selain diterbitkan dalam bentuk cetak, juga telah tersedia dalam bentuk digital yang dapat diunduh di website Jurnal Universitas Sebelas Maret (https://jurnal.uns.ac.id). Artikel ilmiah yang ada di website Jurnal Universitas Sebelas Maret terserbut akan dikelompokkan sesuai dengan bidang ilmunya. Berdasarkan data yang didapatkan, hanya 10 bidang ilmu yang

akan digunakan. Pengelompokan artikel ilmiah pada penelitian ini menerapkan metode klasifikasi *Naive Bayes Classifier*.

2. TEXT MINING

Text mining dapat diartikan sebagai penemuan informasi yang baru dan tidak diketahui sebelumnya oleh komputer, dengan secara otomatis mengekstrak informasi dari sumber-sumber yang berbeda. Kunci dari proses ini adalah menggabungkan informasi yang berhasil diekstraksi dari berbagai sumber[2]. *Text mining* juga dikenal sebagai *Knowledge Discovery from Text* (KDT) atau penemuan pengetahuan dari suatu teks, mengacu pada proses ekstrasi pola yang menarik dari database teks yang sangat besar yang bertujuan untuk menemukan pengetahuan baru[6].

Text mining merupakan penerapan konsep data *mining* yang mengolah data berupa teks. Teknik dalam literatur data *mining* yang sering digunakan dalam *text mining* adalah klasterisasi (*clustering*) dan klasifikasi (*classification*).

Klasterisasi (*clustering*) merupakan proses partisi satu set obyek data ke dalam himpunan bagian yang disebut dengan *cluster*. Obyek yang di dalam *cluster* memiliki kemiripan karakteristik antar satu sama lainnya dan berbeda dengan *cluster* yang lain. [7]

Klasterisasi dapat mengelompokkan data ke dalam kelompok-kelompok tanpa menentukan kategori sebelumnya. Hasil klasterisasi yang baik akan menghasilkan tingkat kesamaan yang tinggi dalam satu *cluster* dan tingkat kesamaan yang rendah antar *cluster*. Kualitas hasil klasterisasi sangat bergantung pada metode yang dipakai. Metode klasterisasi harus dapat mengukur kemampuannya sendiri dalam usaha menemukan suatu pola tersembunyi pada data yang sedang diteliti. Menurut Tan (2011), secara umum metode klasterisasi dapat dibagi menjadi dua, yaitu *hierarchical clustering* dan *partional clustering* [7]

Klasifikasi (*classification*) merupakan penempatan objek-objek ke salah satu dari beberapa kategori yang telah ditetapkan sebelumnya[4]. Data input untuk klasifikasi adalah koleksi record. Setiap record dikenal sebagai instance atau contoh yang ditentukan oleh sebuah *tuple* (x,y). Dimana x adalah himpunan atribut dan y adalah atribut tertentu, yang dinyatakan sebagai label class (juga dikenal sebagai kategori atau atribut target).

Beberapa teknik klasifikasi yang digunakan adalah *decision tree classifier, rulebased* c*lassifier, neural-network, support vector machine*, dan *naïve bayes classifier*. Setiap teknik menggunakan algoritme pembelajaran untuk mengidentifikasi model yang memberikan hubungan yang paling sesuai antara himpunan atribut dan label kelas dari data input.

Menurut Even dan Zohar (2002), *text mining* dibagi menjadi 3 tahap utama, yaitu proses awal terhadap teks (*text preprocessing*), transformasi teks ke dalam bentuk antara (*text transformation*/*feature generation*), dan penemuan pola (*pattern discovery*). Masukan awal dari proses ini adalah suatu data teks dan menghasilkan keluaran berupa pola sebagai hasil interpretasi [4].

3. FEATURE SELECTION

Pada *text clustering* maupun *classification* terdapat suatu permasalahan, yaitu adanya fitur-fitur yang berdimensi tinggi, sehingga diperlukan metode *feature selection* untuk mengurangi dimensi fitur tersebur[8]. *Feature selection* merupakan langkah memilih beberapa fitur (misalnya kata-kata atau istilah) yang akan digunakan ketika melakukan klasifikasi maupun *clustering* . Fitur yang terpilih dapat digunakan untuk mewakili semua fitur yang ada pada dokumen. Salah satu metode *feature selection* yang banyak digunakan dalam *text mining* adalah *document frequency* [9]

Document Frequency (DF) adalah jumlah dokumen yang mengandung suatu *term* tertentu[9].Tiap *term* akan dihitung nilai DF-nya kemudian *term* tersebut diseleksi berdasarkan nilai DF. Jika nilai DF berada di atas atau di bawah nilai *threshold* yang telah ditentukan, maka *term* tersebut akan dibuang[8]. Asumsi awalnya adalah bahwa *term* yang lebih jarang muncul tidak memiliki pengaruh yang besar dalam proses pengelompokan atau pengklasifikasian dokumen. Pembuangan *term* yang jarang ini dapat mengurangi dimensi fitur yang besar pada *text mining*.

Dengan mereduksi fitur yang digunakan dalam proses klasifikasi, akan meningkatkan kinerja klasifikasi. Terdapat 3 syarat suatu data input dinyatakan sangat membantu dalam proses klasifikasi[10], yaitu:

3.1 *Concentration Degree*

Dalam suatu data set dengan berbagai kelas atau kategori, jika fitur kata muncul di satu atau sedikit kelas tapi tidak muncul di kelas lain, fitur kata tersebut memberikan informasi spesifik yang kuat dan sangat membantu dalam proses klasifikasi. Formula yang digunakan untuk menunjukkan rasio seberapa tinggi konsentrasi suatu fitur dalam suatu kelas adalah Persamaan 3.1 di bawah ini:

$$Concentration\ degree = \frac{DF(t,c_i)}{\left(1+\sum_{j=1|j\neq i}^{n} DF(t,c_j)\right)} \quad (3.1)$$

Keterangan:
$DF(t, c_i)$ = jumlah dokumen kelas c_i yang mengandung *term t*
$DF(t, c_j)$ = *jumlah dokumen kelas lain selain* c_i *yang mengandung term t*

3.2 *Disperse Degree*

Jika suatu fitur kata muncul di satu kelas, fitur ini memiliki korelasi yang kuat dengan kelas tersebut. Sehingga, fitur sangat membantu proses klasifikasi apabila tersebar dalam satu kelas. *Disperse degree* menunjukkan tingkat persebaran suatu fitur dalam satu kelas. Sebagai contoh terdapat *m* kelas yang berbeda, C = {c_1, c_2, ..., c_m}, Persamaan 3.2 digunakan untuk mengetahui derajat persebarannya.

$$Disperse\ Degree = \frac{DF(t,c_i)}{N(c_i)} \quad (3.2)$$

Keterangan:
$N(c_i)$ = jumlah dokumen pada kelas c_i

3.3 *Contribusion Degree*

Jika sebuah fitur kata hanya muncul di satu atau sedikit kelas dan kata tersebut tersebar dalam banyak dokumen pada kelas yang sama serta TF atau frekuensi kemunculannya pada satu dokumen tinggi, maka kata tersebut memiliki kontribusi yang tinggi terhadap kelas yang bersangkutan. *Contribusion degree* merupakan penyederhanaan dari *Expectation Crossing Entropy (ECE). Expectation Crossing Entropy (ECE)* adalah salah satu jenis seleksi fitur yang mempertimbangkan TF (*Term Frequemcy*) dan relasi antara *term* / fitur kata dengan kelas. Nilai ECE yang besar menunjukkan bahwa fitur kata tersebut semakin informatif dan membantu proses klasifikasi. Formula ECE yang disederhanakan digunakan untuk menentukan nilai kontribusi suatu fitur terhadap suatu kelas. Formula yang telah disederhanakan ditunjukkan pada Persamaan 3.3.

$$Contribusion\ degree = P(c_i,t) log \frac{p(c_i|t)}{p(c_i)} \quad (3.3)$$

Keterangan:
$P(c_i, t)$ = probabilitas gabungan kelas c_i dan *term* t
$P(c_i|t)$ = probabilitas kelas c_i yang mengandung *term t*
$P(c_i)$ = probabililtas kelas c_i

Berdasarkan 3 prinsip suatu fitur dikatakan sangat membantu dalam proses klasifikasi di atas dan metode *Document Frequency (DF)*, didapatkan metode *feature selection* baru yang disebut *Document Frequency Improved (DFM)*[10]. Persamaan *Document Frequency Improved (DFM)* ditunjukkan pada Persamaan 3.4.

$$DFM(t, c_i) = \frac{DF(t,c_i)}{\left(1+ \sum_{j=1|j\neq i}^{n} DF(t,c_j)\right)} + \frac{DF(t,c_i)}{N(c_i)} + p(c_i, t) log \frac{p(c_i|t)}{p(c_i)} \quad (3.4)$$

4. *NAIVE BAYES CLASSIFIER* (*NBC*)

Metode *Naive Bayes* atau *Naive Bayes Classifier* (*NBC*) adalah salah satu metode yang digunakan untuk klasifikasi teks. *Naive Bayes Classifier* (*NBC*) menggunakan teori probabilitas sebagai dasar teori. Dalam bukunya, Han, J. dan Kamber, M. menyatakan bahwa "*Bayesian* classffiers mempunyai tingkat kecepatan dan akurasi yang tinggi ketika diaplikasikan dalam *database* yang besar" [11]. *Naive Bayes Classifier* adalah metode *classifier* yang berdasarkan probabilitas dan Teorema *Bayesian* dengan asumsi bahwa setiap variabel X bersifat bebas (*independent*). Dengan kata lain, *Naive Bayes Classifier* mengansumsikan bahwa keberadaan sebuah atribut (variabel) tidak ada kaitannya dengan keberadaan atribut yang lain.

Konsep dasar yang digunakan oleh *Naive Bayes* adalah Teorema *Bayes*, yaitu probabilitas $p(C = c_i|D = d_j)$, probabilitas kategori c_i jika diketahui dokumen d_j. Klasifikasi dilakukan untuk menentukan kategori $c \in C$ dari dokumen $d \in D$ dimana $C = \{c_1, c_2, c_3, \dots, c_i\}$ dan $D = \{d_1, d_2, d_3, \dots, d_j\}$. Penentuan dari kategori sebuah dokumen dilakukan dengan mencari nilai maksimum dari $p(C = c_i|D = d_j)$ pada $P = \{p(C = c_i|D = d_j)|c \in C \; dan \; d \in D\}$. Nilai probabilitas $p(C = c_i|D = d_j)$ dapat dihitung dengan Persamaan 4.1.

$$p(C = c_i|D = d_j) = \frac{p(D=d_j|C=c_i) \times p(C=c_i)}{p(D=d_j)} \quad (4.1)$$

dengan $p(C = c_i|D = d_j)$ merupakan nilai probabilitas dari kemunculan dokumen d_j jika diketahui dokumen tersebut berkategori c_i, $p(C = c_i)$ adalah nilai probabilitas kemunculan kategori c_i, dan $p(D = d_j)$ adalah nilai probabilitas kemunculan dokumen d_j.

Naive Bayes menganggap sebuah dokumen sebagai kumpulan dari kata-kata yang menyusun dokumen tersebut, dan tidak memperhatikan urutan kemunculan kata pada dokumen. Sehingga perhitungan probabilitas $p(C = c_i|D = d_j)$ dapat dianggap sebagai hasil perkalian dari probabilitas kemunculan kata-kata pada dokumen d_j. Perhitungan probabilitas $p(C = c_i|D = d_j)$ dapat dituliskan seperti Persamaan 4.2.

$$p(C = c_i|D = d_j) = \frac{\prod_k p(w_k|C = c_i) \times p(C=c_i)}{p(w_1, w_2, w_3, \cdots w_k, \cdots w_n)} \quad (4.2)$$

dengan $\prod_k p(w_k|C = c_i)$ adalah hasil perkalian dari probabilitas kemunculan semua kata pada dokumen d_j.

Proses klasifikasi dilakukan dengan membuat model *probabilistic* dari dokumen training, yaitu dengan menghitung nilai $P(w_k|c)$. Untuk w_{kj} diskrit dengan $w_{kj} \in V = \{v_1, v_2, v_3, \dots, v_m\}$ maka $P(w_k|c)$ dicari untuk seluruh kemungkinan nilai w_{kj} dan didapatkan dengan melakukan perhitungan pada Persamaan 4.3 dan Persamaan 4.4.

$$P(w_k = w_{kj}|c) = \frac{T(w_k=w_{kj.c})}{T(c)} \quad (4.3)$$

$$p(c) = \frac{N(c)}{N} \quad (4.4)$$

dengan $T(w_k = w_{kj.c})$ adalah fungsi yang mengembalikan jumlah kata $w_k = w_{kj}$ pada kategori c, $T(c)$ ad alah fungsi yang mengembalikan jumlah seluruh kata pada kategori c, $N(c)$ adalah fungsi yang mengembalikan jumlah dokumen yang memiliki kategori c , dan N adalah jumlah seluruh dokumen.

Persamaan 4.3 sering dikombinasikan dengan Laplacian Smoothing untuk mencegah persamaan mendapatkan nilai 0, yang dapat mengganggu hasil klasifikasi secara keseluruhan dengan cara menambah nilai 1 pada setiap fitur kata. Sehingga menjadi seperti Persamaan 4.5

$$P(w_k = w_{kj}|c) = \frac{T(w_k=w_{kj.c})+1}{T(c)+|V|} \quad (4.5)$$

dengan $|V|$ merupakan jumlah kemungkinan nilai dari w_{kj}.

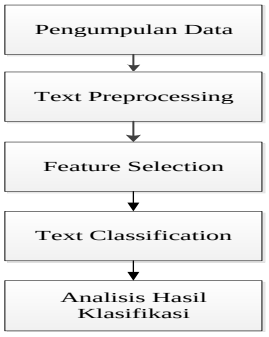
Pemberian kategori dari sebuah dokumen dilakukan dengan memilih nilai c yang memilki nilai $p(C = c_i|D = d_j)$ maksimum, dan dinyatakan dalam Persamaan 4.6.

$$c^* = arg \max_{c \in C} \prod_k p(w_k|C) \times p(c) \quad (4.6)$$

Kategori c^* merupakan kategori yang memiliki nilai $p(C = c_i|D = d_j)$ maksimum. Nilai $p(D = d_j)$ tidak mempengaruhi perbandingan karena untuk setiap kategori nilainya akan sama.

5. METODOLOGI

Metode penelitian pada pengklasifikasian artikel ilmiah ini terdiri dari 5 tahapan, yaitu tahap pengumpulan data, tahap *text preprocessing*, tahap *feature selection*, tahap *text classification*, dan tahap analisis hasil klasifikasi. Tahapan-tahapan tersebut digambarkan seperti Gambar 5.1.



Gambar 5.1 Tahapan Penelitian

5.1 Pengumpulan Data

Pada tahap ini dilakukan pengambilan data yang akan diolah pada penelitian ini yaitu artikel ilmiah yang dipublikasikan oleh Universitas Sebelas Maret (UNS). Data diperoleh dari website Jurnal Universitas Sebelas Maret (https://jurnal.uns.ac.id). Data yang akan digunakan sebagai masukan pada proses klasifikasi adalah bagian judul dan abstrak dokumen yang berbahasa Indonesia.

Sebelum dilakukan klasifikasi, dokumen akan dipilah terlebih dahulu. Dokumen dengan abstrak yang tidak berbahasa Indonesia atau memiliki keterangan tidak lengkap (tidak ada judul atau abstrak) akan dihapus.

5.2 *Text* Preprocessing

Pada tahap *preprocessing*, awal mula data mentah dilakukan proses *case folding*, proses ini akan menghilangkan karakter selain huruf abjad dan mengubah semua huruf menjadi huruf kecil (*lowercase*). Data hasil *case folding* yang masih berupa kalimat akan dipotong berdasarkan tiap kata penyusunnya. Proses ini disebut dengan tokenisasi. Hasil dari proses tokenisasi ini menghasilkan fitur yang digunakan sebagai data pembelajaran mesin oleh *NBC*. Selanjutnya kata-kata yang dianggap tidak penting atau tidak menggambarkan isi dokumen (*stopword list*) akan dihapus. *Stopword list* atau *stoplist* berbahasa Indonesia yang digunakan didapatkan dari penelitian yang dilakukan oleh Fadillah Z Tala (2003) [12] dengan 13 kata tambahan: abstrak, abstract, penelitian, makalah, penulis, tujuan, bertujuan, membahas, permasalahan, dibahas, artikel, dll, dan memiliki.

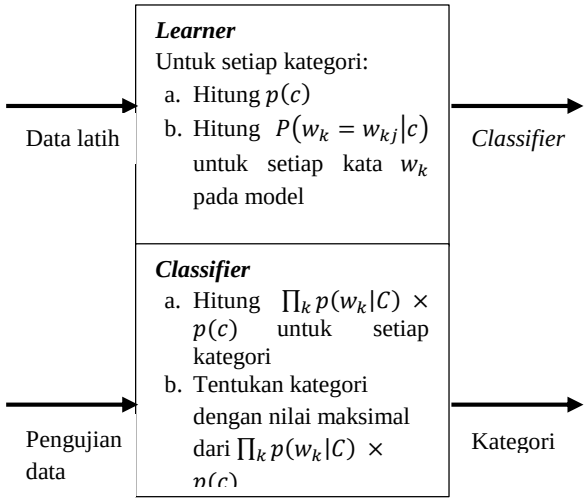
Kata-kata yang tidak termasuk dalam *stoplist* kemudian akan diubah ke dalam bentuk dasarnya (*root word*) dengan menghilangkan imbuhan awalan dan akhirannya. Algoritma stemming yang digunakan pada penelitian ini adalah algoritma Nazief & Adriani [13].

5.3 Feature Selection

Nilai *Document Frequency Improved* (*DFM*) dicari menggunakan inputan hasil proses *text transformation* dengan cara menghitung banyaknya dokumen yang mengandung *term t*, kemudian menghitung nilai *concentration degree*, *disperse degree*, dan *contribution degree*. Nilai DFM ini akan digunakan sebagai pertimbangan apakah kata akan digunakan sebagai fitur pada saat proses klasifikasi atau dihilangkan. Semakin besar nilai DFM maka fitur semakin dianggap penting. Sebuah nilai *threshold* dipilih untuk mengurangi fitur yang tidak diperlukan. Beberapa nilai *threshold* mulai dari 1 diterapkan dan dibandingkan masing-masing rata-rata akurasi hingga ditemukan nilai *threshold* yang paling optimal dengan nilai rata-rata akurasi paling tinggi.

5.4 *Text Classification*

Klasifikasi menggunakan *Naive Bayes* terbagi menjadi dua tahapan yang disajikan pada Gambar 5.2 berikut:



Gambar 5.2 Tahapan klasifikasi dengan NBC

Tahap *learning* atau pembelajaran ini akan menciptakan pengetahuan awal sebagai dasar dalam menentukan klasifikasi dengan menghitung $p(c)$, probabilitas pada setiap kategori. Setelah itu menentukan $p(w_k|C)$, frekuensi setiap kata w_k pada setiap kategori.

Pengklasifikasian ditentukan dari nilai terbesar perhitungan $\prod_k p(w_k|C) \times p(c)$ untuk setiap kategori.

5.5 Analisis Hasil Klasifikasi

Pada tahap ini akan dihitung keakuratan hasil klasifikasi yang telah dilakukan. Selain itu juga akan dilakukan perhitungan *precision*, *recall*, dan *f-measure*. Hasil klasifikasi disajikan menggunakan *confusion matrix*. Tabel 5.1 menunjukkan penggunaan *confusion matrix*

Tabel 5.1 Confusion matrix

Kelas Sebenarnya	Kelas Hasil Klasifikasi					
	C_1	C_2	C_3	...	C_n	
C_1	TP_C_1	Error	Error	...	Error	Total_C_1
C_2	Error	TP_C_2	Error	...	Error	Total_C_2
C_3	Error	Error	TP_C_3	...	Error	Total_C_3
⋮	:	:	:	⋮	:	⋮
C_n	Error	Error	Error	...	TP_C_n	Total_C_n
	Terprediksi_C_1	Terprediksi_C_2	Terprediksi_C_3	...	Terprediksi_C_n	

Nilai akurasi secara keseluruhan didapatkan dari perhitungan jumlah prediksi benar yang sesuai (TP atau *True Positive* dibagi dengan jumlah seluruh data set atau jumlah prediksi benar yang sesuai (TP atau *True Positive*) ditambahkan dengan jumlah data yang *error* [14]. Rumus perhitungannya dapat dinyatakan dengan Persamaan 5.1. TP atau *True Positive* menunjukkan data yang diklasifikasi sesuai dengan kelas sebenarnya, sedangkan *Error* menunjukkan data yang diklasifikasikan tidak sesuai dengan kelas sebenarnya. Berdasarkan *confusion matrix* kemudian dilakukan penghitungan *precision*, *recall*, dan *f-measure*. *Precision* (P) adalah jumlah sampel berkategori positif diklasifikasikan benar dibagi dengan total sampel yang diklasifikasikan sebagai sampel positif atau *classified positif* atau jumlah seluruh hasil klasifikasi (Terprediksi_*Ci*). Formula perhitungan nilai *precision* dapat dilihat pada Persamaan 5.2. *Recall* (R) adalah jumlah hasil klasifikasi yang bernilai benar dibagi dengan total sampel dalam *testing set* berkategori positif atau *actual positif* atau jumlah nilai benar yang seharusnya (Total_*Ci*). Formula perhitungan nilai *recall* dapat dilihat pada Persamaan 5.3. Sementara *F-measure* (F) adalah nilai akurasi matriks yang menghitung rasio dari hasil yang benar dan berlaku sebagai nilai ratarata harmonis dari *precision* dan *recall*. Formula perhitungan nilai *F-measure* dapat dilihat pada Persamaan 5.4 .

$$accuration = \frac{TP(c_1)+TP(c_2)+TP(c_3)+\cdots+TP(c_n)}{Total(c_1)+Total(c_2)+Total(c_3)+\cdots+Total(c_n)} \quad (5.1)$$

$$p(c_i) = \frac{TP(c_i)}{Terprediksi(c_i)} \quad (5.2)$$

$$r(c_i) = \frac{TP(c_i)}{Total(c_i)} \quad (5.3)$$

$$F(c_i) = \frac{2p(c_i) * r(c_i)}{p(c_i)+r(c_i)} \quad (5.4)$$

Keterangan:
$p(c_i)$ = nilai precision pada kelas c_i

$r(c_i)$ = nilai *recall* pada kelas c_i
$F(c_i)$ = nilai f-measure pada kelas c_i

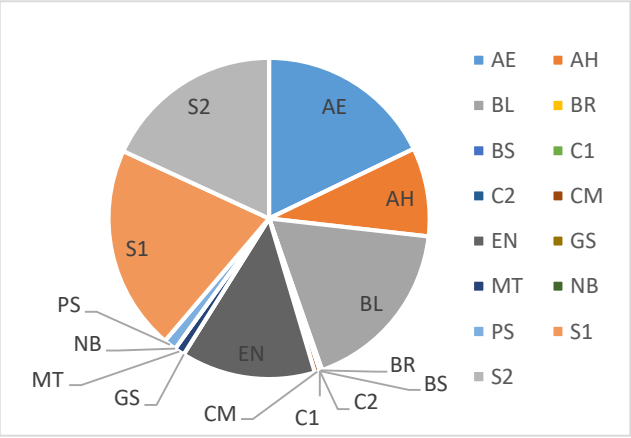
6. HASIL DAN PEMBAHASAN

6.1 Pengumpulan Data

Pengambilan data dilakukan dengan mengambil artikel ilmiah dari website Jurnal Universitas Sebelas Maret (https://jurnal.uns.ac.id). Artikel ilmiah yang diambil adalah artikel ilmiah dengan abstrak berbahasa Indonesia. Jumlah data yang didapatkan sebanyak 392 artikel yang dipublikasikan pada tahun 2006 sampai dengan 2017. Artikel-artikel ilmiah ini kemudian dikelompokkan sesuai dengan bidang ilmunya. Pengelompokan data pada tahap ini sebenarnya dapat menggunakan kata kunci yang biasanya terdapat pada abstrak masing-masing artikel, akan tetapi berdasarkan data yang didapatkan, banyak artikel tidak mencantumkan kata kunci pada abstraknya, sehingga pengelompokan dilakukan secara manual. Hasil pengelompokan ini nantinya akan digunakan sebagai label pada data training untuk menentukan model pembelajaran dan sebagai variabel pembanding untuk menghitung *precision*, *recall*, *f-measure*, dan akurasinya. Tabel 6.1 menampilkan jumlah data didapat yang telah dikelompokkan sesuai bidang ilmunya.

Tabel 6.1 Jumlah data yang didapat

Bidang Ilmu	Kode	Jumlah
Agriculture & Environment	AE	70
Biology	BL	70
Biosciences	BS	0
Biomedical Research	BR	1
Clinical and Experimental Medicine I	C1	0
Clinical and Experimental Medicine II	C2	0
Neuroscience & Behavior	NB	0
Chemistry	CM	2
Physic	PS	5
Geosciences & Space Sciences	GS	0
Engineering	EN	53
Mathematics	MT	4
Social Science I	S1	81
Social Science II	S2	71
Arts & Humanities	AH	35



Gambar 6.1 Distribusi data artikel ilmiah

Pada Gambar 6.1 distribusi data artikel ilmiah yang didapatkan di atas dapat dilihat bahwa hanya 6 kelas yang terlihat menonjol. Hal ini dikarenakan 9 kelas lainnya memiliki data sangat kecil atau bahkan tidak memiliki data. Kelas-kelas yang tidak memiliki data tidak akan muncul pada hasil klasifikasi. Kelas-kelas ini akan muncul sebagai kelas hasil klasifikasi hanya jika ada data berlabel kelas bersangkutan dalam data latih. Oleh karena itu kelas-kelas yang tidak memiliki data ini tidak akan digunakan dalam penelitian ini.

Kelas-kelas yang memiliki data sangat kecil akan digabungkan ke dalam kelas yang memiliki kemiripan pada ruang lingkup kajiannya. Data kelas *Biomedical Research* digabungkan dengan kelas *Biology* menjadi kelas *Biology & Biomedical Research*, data kelas *Mathematics* dan kelas *Engineering* dijadikan satu menjadi kelas *Engineering & Mathematics*, serta kelas *Physic* dan *Chemistry* digabungkan menjadi satu kelas membentuk kelas *Physic & Chemistry*. Data akhir yang akan digunakan pada penelitian ini dapat dilihat pada Tabel 6.2.

Tabel 6.2 Jumlah data yang digunakan

Bidang Ilmu	Kode	Jumlah
Agriculture & Environment	AE	70
Arts & Humanities	AH	35
Biology & Biomedical Research	BB	71
Engineering & Mathematics	EM	57
Physic & Chemistry	PC	7
Social Science I	S1	81
Social Science II	S2	71

Jumlah artikel ilmiah yang digunakan sebanyak 392 data. Data ini kemudian dibagi menjadi data training dan data test masing-masing sebanyak 292 dan 100 data dengan komposisi data masing-masing kelas ditampilkan pada Tabel 6.3.

Tabel 6.3 Komposisi data

Kelas	Jumlah Dokumen Training	Jumlah Dokumen Testing
AE	52	18
AH	26	9
BB	53	18
EM	43	14
PC	5	2
S1	60	21
S2	53	18
	292	**100**

Pada Tabel 6.3 dapat dilihat bahwa data training pada kelas *Arts & Humanities (AH)* dan *Physic & Chemistry (PC)* memiki jumlah data training paling sedikit dan selisih yang cukup banyak dibandingkan kelas lain. Oleh karena itu, agar jumlah data training tiap kelas menjadi lebih seimbang, maka untuk data training pada kelas *Arts & Humanities* (AH) akan gandakan 2 kali sedangkan pada kelas *Physic & Chemistry* (PC) akan gandakan 10 kali.

6.2 *Text* Preprocessing

Tahap *text preprocessing* terdiri dari beberapa proses, yaitu menghilangkan karakter selain huruf abjad dan mengubah semua huruf menjadi huruf kecil (*case folding*), memotong kalimat berdasarkan tiap kata penyusunnya (tokenisasi), menghapus kata-kata yang dianggap tidak penting (*stoplist*), mengubah kata-kata ke dalam bentuk dasarnya (*stemming*). Contoh hasil tahapan *text preprocessing* ditunjukkan dalam Tabel 6.4.

Tabel 6.4 Contoh Hasil *Text Preprocessing*

Judul	PELAKSANAAN KEWENANGAN KHUSUS PEMERINTAHAN MENURUT UU NOMOR 11 TAHUN 2006 TENTANG PEMERINTAHAN ACEH (SUATU PENELITIAN DI KABUPATEN ACEH BARAT)
Abstrak	Abstract Abstrak Penelitian ini merupakan penelitian hukum empiris (empirical research) dengan menggunakan pendekatan perundang-undangan (statute approach). Penelitian bersifat deskriptif analitis. Data primer dan data sekunder yang digunakan dalam penelitian ini dianalisis secara kualitatif. Hasil penelitian menunjukkan bahwa dalam rangka melaksanakan urusan wajib lainnya yang menjadi kewenangan khusus sebagaimana diamanatkan dalam Undang-Undang Nomor 11 Tahun 2006 tentang Pemerintahan Aceh, Pemerintah Kabupaten Aceh Barat telah melaksanakan keempat bidang yang menjadi keistimewaan Aceh. Namun dalam pelaksanaannya menghadapi berbagai hambatan. Adapun hambatan-hambatan dan dukungan elit daerah Kata Kunci: Kewenangan Khusus,Pemerintah Aceh, Pemerintahan Aceh
Text Preprocessing	laksana wenang khusus pemerintah uu nomor pemerintah aceh kabupaten aceh barat hukum empiris empirical dekat undang undang statute approach sifat deskriptif analitis data primer data sekunder analisis kualitatif hasil rangka laksana urus wajib wenang khusus amanat undang undang nomor pemerintah aceh pemerintah kabupaten aceh barat laksana empat bidang istimewa aceh laksana hadap hambat hambat hambat dukung elit daerah kunci wenang khusus pemerintah aceh pemerintah aceh

6.3 Feature Selection

Penerapan *feature selection* diawali dengan menghitung nilai *Document Frequency Improved (DFM)* untuk setiap kata yang ada di dalam data training menggunakan Persamaan 3.4. Hasil yang diperoleh kemudian dimasukkan ke dalam *database.*

Tahap selanjutnya yaitu menentukan besar *threshold* untuk fitur yang diseleksi. Penentuan *threshold* dilakukan beberapa kali sehingga ditemukan hasil paling optimal. Setelah ditentukan, fitur – fitur dengan nilai di bawah batas/*threshold* akan dihapus. Fitur-fitur yang telah lolos seleksi akan digunakan sebagai fitur dalam proses klasifikasi dengan *Naive Bayes*. Contoh hasil seleksi fitur dapat dilihat pada Tabel 6.5.

Tabel 6.5 Contoh hasil *feature selection*

Judul	PELAKSANAAN KEWENANGAN KHUSUS PEMERINTAHAN MENURUT UU NOMOR 11 TAHUN 2006 TENTANG PEMERINTAHAN ACEH (SUATU PENELITIAN DI KABUPATEN ACEH BARAT)
Abstrak	Abstract Abstrak Penelitian ini merupakan penelitian hukum empiris (empirical research) dengan menggunakan pendekatan perundang-undangan (statute approach). Penelitian bersifat deskriptif analitis. Data primer dan data sekunder yang digunakan dalam penelitian ini dianalisis secara kualitatif. Hasil penelitian menunjukkan bahwa dalam rangka melaksanakan urusan wajib lainnya yang menjadi kewenangan khusus sebagaimana diamanatkan dalam Undang-Undang Nomor 11 Tahun 2006 tentang Pemerintahan Aceh, Pemerintah Kabupaten Aceh Barat telah melaksanakan keempat bidang yang menjadi keistimewaan Aceh. Namun dalam pelaksanaannya menghadapi berbagai hambatan. Adapun hambatan-hambatan dan dukungan elit daerah Kata Kunci: Kewenangan Khusus,Pemerintah Aceh, Pemerintahan Aceh
Text Preprocessing	laksana wenang khusus pemerintah uu nomor pemerintah aceh kabupaten aceh barat hukum empiris empirical dekat undang undang statute approach sifat deskriptif analitis data primer data sekunder analisis kualitatif hasil rangka laksana urus wajib wenang khusus amanat undang undang nomor pemerintah aceh pemerintah kabupaten aceh barat laksana empat bidang istimewa aceh laksana hadap hambat hambat hambat dukung elit daerah kunci wenang khusus pemerintah aceh pemerintah aceh
Hasil *Feature Selection*	wenang - uu - nomor - hukum - empiris - undang - undang - sifat - analisis - hasil - wenang - undang - undang - nomor - elit - kunci - wenang -

6.4 *Text Classification*

Klasifikasi dilakukan dengan menghitung nilai probabilitas berdasarkan fitur-fitur yang telah terseleksi sebelumnya. Nilai fitur yang digunakan dalam klasifikasi ini adalah nilai *Term Frequency (TF).* Setiap *term* pada data latih yang sesuai dengan fitur dihitung nilai TF-nya kemudian dicari nilai probabilitas setiap fiturnya dan probabilitas kelas bersangkutan.

Setiap *term* pada data uji yang sesuai dengan fitur dihitung probabilitas kemunculannya dalam kelas kemudian dikalikan dengan probabilitas kelasnya. Setelah semua kemungkinan kelas yang memiliki relasi dengan fitur dihitung probabilitasnya, semua nilainya dibandingkan dan dicari nilai maksimalnya. Kelas yang memiliki nilai terbesar itulah yang akan terpilih.

Hasil klasifikasi disajikan menggunakan *confusion matrix*. Contoh *confusion matrix* dapat dilihat pada Tabel 6.6 dengan hasil akurasi, *precision, recall,* dan *f-measure*-nya pada Tabel 6.7.

Tabel 6.6 *Confusion matrix* tanpa *feature selection* percobaan ke-1

Kelas Sebenarnya	Kelas Hasil Klasifikasi							
	AE	AH	BB	EM	PC	S1	S2	
AE	18	0	0	0	0	0	0	18
AH	2	7	0	0	0	0	0	9
BB	15	0	3	0	0	0	0	18
EM	14	0	0	0	0	0	0	14
PC	0	1	0	0	0	1	0	2
S1	15	0	0	0	0	6	0	21
S2	9	0	0	0	0	0	9	18
	73	8	3	0	0	7	9	43

Tabel 6.7 Akurasi, *precision, recall,* dan *f-measure* tanpa *feature selection* percobaan ke-1

Kelas	Akurasi	*Precision*	*Recall*	*F-measure*
AE	0,43	0,247	1,000	0,396
AH		0,875	0,778	0,824
BB		1,000	0,167	0,286
EM		0,000	0,000	0,000
PC		0,000	0,000	0,000
S1		0,857	0,286	0,429
S2		1,000	0,500	0,667
	0,43	0,568	0,389	0,301

6.5 Analisis Hasil Klasifikasi

Artikel ilmiah yang telah diklasifikasikan akan dianalisis hasilnya dengan menghitung nilai akurasi, *precision*, *recall*, dan *f-measure*. Perhitungan keempat metode ini dilakukan pada setiap percobaan dengan *threshold* yang berbeda. Masing-masing *threshold* diujikan sebanyak 5 kali dengan setiap percobaan menggunakan data training dan data uji yang telah diacak sebelumnya.

Pengujian pertama dilakukan tanpa menerapkan *feature selection*. Hasil rata-rata akurasi tanpa *feature selection* dapat dilihat pada Tabel 6.8.

Tabel 6.8 Hasil pengujian tanpa feature selection

Percobaan ke-	Akurasi	*Precision*	*Recall*	*F-measure*
1	43,0%	56,8%	38,9%	30,1%
2	46,0%	65,6%	39,6%	39,4%
3	45,0%	67,9%	39,7%	39,9%
4	44,0%	73,8%	38,4%	38,7%
5	36,0%	72,2%	32,3%	31,2%
Rata-rata	42,8%	67,3%	37,8%	35,9%

Penerapan nilai *threshold* dilakukan dengan meningkatkan nilainya secara bertahap hingga didapatkan nilai rata-rata akurasi paling tinggi. Pada Tabel 6.9 menampilkan hasil akurasi, *precision, recall,* dan *f-measure* kelima percobaan pada nilai *threshold* 1.

Tabel 6.9 Hasil pengujian pada *threshold* 1

Percobaan ke-	Akurasi	*Precision*	*Recall*	*F-measure*
1	80,0%	73,1%	67,9%	66,3%
2	83,0%	74,2%	66,0%	64,8%
3	79,0%	70,1%	59,7%	59,2%
4	81,0%	73,7%	67,1%	67,2%
5	77,0%	71,2%	67,5%	65,2%
Rata-rata	80,0%	72,4%	65,7%	64,5%

Pada pengujian nilai *threshold* 1,5 menghasilkan rata-rata akurasi yang meningkat sebanyak 7,4% seperti yang ditampilkan pada Tabel 6.10.

Tabel 6.10 Hasil pengujian pada *threshold* 1,5

Percobaan ke-	Akurasi	*Precision*	*Recall*	*F-measure*
1	88,0%	76,5%	77,0%	76,6%
2	86,0%	75,2%	72,8%	73,0%
3	87,0%	74,7%	74,6%	74,4%
4	85,0%	74,0%	73,2%	73,3%
5	91,0%	78,5%	80,0%	79,0%
Rata-rata	87,4%	75,8%	75,5%	75,3%

Nilai *threshold* kemudian ditingkatkan lagi menjadi 2. Hasil rata-rata akurasi pada *threshold* ini dapat dilihat pada Tabel 6.11.

Tabel 6.11 Hasil pengujian pada *threshold* 2

Percobaan ke-	Akurasi	*Precision*	*Recall*	*F-measure*
1	88,0%	76,5%	76,9%	76,6%
2	86,0%	76,6%	73,6%	74,2%
3	87,0%	75,2%	74,6%	74,6%
4	89,0%	77,3%	77,4%	77,2%
5	89,0%	77,2%	78,5%	77,5%
Rata-rata	87,8%	76,6%	76,2%	76,0%

Pada Tabel 6.11 dapat dilihat bahwa rata-rata akurasi masih meningkat sehingga nilai *threshold* ditingkatkan lagi menjadi 2,5. Hasil pengujian pada *threshold* ini ditampilkan pada Tabel 6.12.

Tabel 6.12 Hasil pengujian pada *threshold* 2,5

Percobaan ke-	Akurasi	*Precision*	*Recall*	*F-measure*
1	81,0%	70,1%	70,1%	69,9%
2	81,0%	68,8%	69,4%	68,9%
3	85,0%	72,6%	72,6%	72,4%
4	87,0%	88,6%	80,5%	82,9%
5	88,0%	76,0%	76,5%	75,7%
Rata-rata	84,4%	75,2%	73,8%	74,0%

Pada Tabel 6.12 dapat dilihat bahwa rata-rata akurasi menurun sebanyak 3,4%, dari 87,8% menjadi 84,4%, sehingga nilai *threshold* tidak lagi ditingkatkan.

Tabel 6.13 Rata-rata akurasi, *precision*, *recall*, dan *f-measure* setiap *threshold*

Feature Selection	Akurasi	Precision	Recall	F-measure
Tanpa Feature Selection	42,8%	67,3%	37,8%	35,9%
Threshold 1	80,0%	72,4%	65,7%	64,5%
Threshold 1.5	87,4%	75,8%	75,5%	75,3%
Threshold 2	87,8%	76,6%	76,2%	76,0%
Threshold 2.5	84,4%	75,2%	73,8%	74,0%

Berdasarkan Tabel 6.13 dapat dilihat perbedaan nilai yang dihasilkan sebelum dan sesudah diterapkan *feature selection*. Pengujian sebelum *feature selection* diterapkan menghasilkan rata-rata akurasi, *precision, recall,* dan *f-measure* terendah dengan nilai masing-masing sebesar 42,8%, 67,3%, 37,8%, dan 35,9%. Pengujian setelah menerapkan *feature selection* mengalami peningkatan pada rata-rata akurasinya.

Pada *threshold 1* hingga *threshold* 2 rata-rata akurasi terus mengalami peningkatan namun pada nilai *threshold* 2,5 rata-rata akurasi menurun. Hasil terbaik ditunjukkan pada nilai *threshold* 2 dengan rata-rata akurasi, *precision, recall,* dan *f-measure* masing-masing sebesar 87,8%, 76,6%, 76,2%, dan 76,0%.

7. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan dapat disimpulkan bahwa pengelompokkan artikel ilmiah menggunakan metode Naive Bayes Classifier (NBC) berhasil dilakukan yang ditunjukkan melalui evaluasi akurasi, *precision, recall,* dan *f-measure*.

Pengujian tanpa menerapkan *feature selection* menghasilkan rata-rata akurasi, *precision, recall,* dan *f-measure* terendah dengan nilai masing-masing 42,8%, 67,3%, 37,8%, dan 35,9%. Namun setelah diterapkan *feature selection* hasilnya mulai meningkat dengan hasil terbaik ditunjukkan pada *threshold* 2 dengan nilai akurasi, *precision, recall,* dan *f-measure* masing-masing sebesar 87,8%, 76,6%, 76,2%, dan 76,0%.

8. DAFTAR PUSTAKA

[1] W. Glänzel and A. Schubert, "A new classification scheme of science fields and subfields designed for scientometric evaluation purposes," no. July 2015, 2003.

[2] M. Hearst, "What is Text Mining?," 2003. [Online]. Available: http://people.ischool.berkeley.edu/~hearst/text-mining.html. [Accessed: 24-Apr-2016].

[3] V. Suryaningsih, "CLUSTERING DOKUMEN MENGGUNAKAN ALGORITMA SELF-ORGANIZING MAP (SOM) (STUDI KASUS : DOKUMEN SKRIPSI DI FAKULTAS PERTANIAN UNS)," pp. 1–11, 2015.

[4] D. Yanti, "Analisis akurasi algoritma," Universitas Sumatera Utara, 2013.

[5] S. Natalius, "Metoda Naïve Bayes Classifier dan Penggunaannya pada Klasifikasi Dokumen," *J. Inst. Teknol. Bandung*, no. 3, 2011.

[6] L. Maimon, Oded; Rokach, *Data Mining and Knowledge Discovery Handbook*. Springer Science, 2006.

[7] M. Irwansyah, Edy; Faisal, *Advamced Clustering: Teori dan Aplikasi*. Yogyakarta: DeePublish, 2015.

[8] D. P. Langgeni, Z. K. A. Baizal, and Y. F. A. W, "Clustering Artikel Berita Berbahasa Indonesia Menggunakan Unsupervised Feature Selection," *Semin. Nas. Inform. 2010*, vol. 2010, no. semnasIF, pp. 1–10, 2010.

[9] Ø. L. Garnes, "Feature Selection For Text Categorisation," Trondheim, 2009.

[10] W. Zheng and G. Feng, "Feature Selection Method Based on Improved Document Frequency," *TELKOMNIKA*, vol. 12, no. 4, pp. 905–910, 2014.

[11] A. R. Indranandita, Amalia; Susanto, Budi; C., "Sistem Klasifikasi dan Pencarian Jurnal dengan Menggunakan Metode Naive Bayes dan Vector Space Model," *J. Inform.*, vol. 4, no. 2, p. 10, 2008.

[12] F. Z. Tala, "A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia," 2003.

[13] B. Nazief, M. Adriani, J. Asian, S. M. M. TAHAGHOGHI, and H. E. Williams, "Stemming Indonesian : A Confi x-Stripping Approach," vol. 6, no. 4, pp. 1–33, 2007.

[14] D. M. W. Powers, "Evaluation : From Precision , Recall and F-Factor to ROC , Informedness , Markedness & Correlation," no. December, 2007.