JURNAL INFORMATIKA, Vol.4 No.1 April 2017, pp. 38~45
ISSN: 2355-6579
E-ISSN**:** 2528-2247

Penerapan *PSO* Untuk Seleksi Fitur Pada Klasifikasi Dokumen Berita Menggunakan *NBC*

Erfian Junianto[1], Dwiza Riana[2]

[1]Universitas BSI
email: erfian.ejn@bsi.ac.id

[2]STMIK Nusa Mandiri Jakarta
email: dwiza@nusamandiri.ac.id

Abstrak

Digitalisasi informasi membuat penyebaran informasi menjadi lebih cepat, aktual, dan murah. Informasi yang disebarkan tersebut terjadi dalam bentuk teks, yang mana banyak informasi yang terkandung di dalamnya. Karena banyaknya informasi penting yang terkandung di dalam dokumen teks (berita), maka dibutuhkan metode tertentu untuk menklasifikasikannya. Beberapa penelitian telah dilakukan, namum belum ada yang menerapkan Particle Swarm Optimization (PSO) untuk seleksi fitur pada klasifikasi dokumen. Maka, dalam penelitian ini akan diterapkan PSO untuk melakukan seleksi fitur, dan juga Naïve Bayes Classifier (NBC) untuk klasifikasinya. Data yang digunakan berasal dari 20 Newsgroups. Model percobaan membagi dokumen training dari 10% hingga 90%. Hal ini dilakukan untuk mengetahui model mana yang akan menghasilkan akurasi tertinggi. Dari percobaan dengan model tersebut diketahui, akurasi tertinggi yang dicapai adalah 85,42% dengan dokumen training sebesar 80% (15.077 dokumen). Sedangkan, percobaan menggunakan contoh dokumen yang berbeda, dengan kelas yang sudah ditentukan menghasilkan akurasi hingga 99,87%. Dokumen testing yang digunakan sebesar 20% (3.770 dokumen).

Kata Kunci*: Particle Swarm Optimization, Naïve Bayes Classifier, Klasifikasi Dokumen, Akurasi, Text Mining.*

Abstract

Information digitization makes information dissemination faster, actual, and cheaper. The information disseminated occurs in the form of text, which contains much of the information contained in it. Because of the vast amount of important information contained in text documents (news), it takes certain methods to classify them. Several studies have been conducted, but none have implemented Particle Swarm Optimization (PSO) for feature selection on document classification. So, in this research will be applied PSO to perform feature selection, and also Naïve Bayes Classifier (NBC) for its classification. The data used comes from 20 Newsgroups. The trial model divides training documents from 10% to 90%. This is done to find out which model will produce the highest accuracy. From the experiments with the model is known, the highest accuracy achieved is 85.42% with training documents of 80% (15,077 documents). Meanwhile, experiments using different document samples, with a predetermined class yielding accuracy of up to 99.87%. Test document used is 20% (3770 documents).

Keywords*: Particle Swarm Optimization, Naïve Bayes Classifier, Document Classification, Accuracy, Text Mining.*

1. Pendahuluan

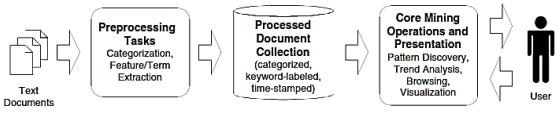
Kehadiran Komputer *personal* dan perkembangan internet membuat digitalisasi informasi semakin mudah. Berdasarkan data yang dikumpulkan oleh badan sensus Amerika Serikat, pada januari 2014 tingkat penetrasi internet di dunia mencapai angka 2,4 miliar atau sekitar 35% (Andri, 2014).

Penggunaan situs memungkinkan penyebaran lebih cepat, aktual, murah, dan ramah lingkungan (Andri, 2014). data digital tersaji dalam bentuk teks, yang mana merupakan penyusun dokumen yang tidak terstruktur dan tidak ada persyaratan khusus untuk menyusunnya (Weiss, Indurkhya, Zhang, & Damerau, 2005). Semakin banyak

Diterima Januari 17, 2017; Revisi Januari 26, 2017; Disetujui Maret 15, 2017

berita, semakin banyak pula dokumen digital yang terkumpul, sehingga semakin sulit bagi pembaca untuk memilih berita yang sesuai dengan keinginan.

Text mining dapat didefinisikan sebagai suatu proses menggali informasi dimana seorang pengguna berinteraksi dengan sekumpulan dokumen menggunakan *tools* analisis, yang merupakan komponen-komponen dalam *data mining* salah satunya adalah klasifikasi. Tujuan dari *text mining* adalah untuk mendapatkan informasi yang berguna dari sekumpulan dokumen. Maka dari itu, sumber data yang digunakan pada *text mining* adalah kumpulan teks yang memiliki format yang tidak terstruktur atau minimal semi terstruktur. Adapun tugas khusus dari *text mining* antara lain yaitu pengkategorisasian teks *(text categorization)* dan pengelompokan teks *(text clustering)* (Feldman & Sanger, 2007, hal. 1). Gambar 1. merupakan gambaran umum tentang kerangka *text mining*.



Gambar 1. *High-level text mining functional architecture*

Sumber: (Feldman & Sanger, 2007, hal. 15)

Klasifikasi terhadap dokumen teks dengan metode *text mining*, akan mempermudah dalam penemuan topik berita yang diingikan. Banyak metode yang diterapkan untuk melakukan klasifikasi terhadap dokumen teks. Diantaranya, Klasifikasi berita berbahasa Indonesia menggunakan *Naïve Bayes Classifier* (Wibisono, 2005), klasifikasi dokumen teks berbahasa Indonesia menggunakan Naïve Bayes (Samodra, Sumpeno, & Hariadi, 2009), klasifikasi emosi untuk teks berbahasa Indonesia (Destuardi & Sumpeno, 2009), klasifikasi dokumen berita, dan abstrak akademis (Wibisono, 2005; Hamzah, 2012). Pada dasarnya klasifikasi tersebut melibatkan penerapan teknik seperti *Information Retreival* (IR), *Natural Language Processing* (NLP), *Data Mining* (DM), *Information Extraction* (IE). Berbagai tahapan proses tersebut dapat dikombinasikan ke dalam alur kerja tunggal. (Ghosh, Roy, & Bandyopadhyay, 2012).

Tahap yang sangat penting sebelum klasifikasi adalah seleksi fitur (*feature selection*). Tahap ini sangatlah berpengaruh pada klasifikasi (Tu, Chuang, Chang, & Yang, 2007), dimana pada tahap ini akan dibuang kata yang menjadi fitur namun tidak relevan atau yang terjadi *redudant*. Jumlah fitur yang besar akan mengakibatkan "kutukan dimensi", yang merupakan masalah besar dalam klasifikasi (Xue, Zhang, & Browne, 2012). Untuk mengatasi hal tersebut, akan diterapkan metode *Particle Swarm Optimization* (PSO). Kelebihan menerapkan metode ini adalah karakteristik PSO yang menerapkan perilaku sosial dari binatang. Seperti sekumpulan burung dalam suatu gerombolan.

PSO terdiri dari sekumpulan partikel yang mencari posisi terbaik, yang merupakan posisi terbaik untuk masalah optimasi dalam ruang fitur. Penerapan PSO sebagai penentu parameter regulasi akan memberikan pengaruh pada *accuracy* klasifikasi (Widiasri, Justitia, & Arifin, 2011). Penerapan PSO pada *data mining* sebagai langkah untuk mendeteksi kerusakan pada perangkat lunak juga memberikan hasil yang efektif, walaupun masih kurang menunjukkan kinerja jika diterapkan pada SVM (Wahono & Suryana, 2013). PSO juga mampu meningkatkan *precision* dari *term* extraction (Syafrullah & Salim, 2010). Kemudian penelitian yang menerapkan PSO sebagai metode *seleksi fitur* dan NBC sebagai klasifikasi, menghasilkan *accuracy* sebesar 76,08% dengan data pegawai dari RIG Tenders Indonesia.

Setelah dilakukan *seleksi fitur*, maka dilakukan klasifikasi yang akan melalui tahapan dimana kumpulan dokumen diidentifikasi berdasarkan inti dari isi dokumen teks tersebut. Metode yang digunakan adalah *Naïve Bayes Classifier* (NBC). Metode NBC dipilih karena kesederhanaan dan kecepatan komputasinya namun memiliki akurasi yang tinggi (Wibisono, 2005; Korde & Mahender, 2012). Metode NBC juga memiliki kinerja yang baik terhadap pengklasifikasian data dokumen yang mengandung angka maupun teks. Sebelum tahap klasifikasi, dokumen harus direpresentasikan menjadi *vector*. Hal ini dilakukan karena *classification algorithm* tidak bisa memproses dokumen secara langsung. Metode yang sering digunakan adalah *Term Frequency-Inverse Document Frequency* (TF-IDF) (Polettini, 2004; Dalal & Zaveri, 2011; Dewi & Supriyanto, 2013).

Data yang akan digunakan untuk penelitian adalah *20 Newsgroups,* dengan 20 kategori dan total dokumen sebanyak 18.846. Banyak penelitian yang menggunakan *datasets* ini, diantaranya

penelitian yang menggunakan *Bayes Formula* sebagai *document preprocessing*. Penelitiannya menggunaan tiga metode yaitu *Naïve Bayes*, *Naïve Bayes-Support Vector Machine* (*Hybrid*), dan *Naïve Bayes-Self Organizing Map* (*Hybrid*). Hasilnya adalah, *accuracy* sebesar 77,82%, 79,55%, dan 34% (Isa, Hong, Kallimani, & Rajkumar, 2008). Nilai *accuracy* 77,82% masih mungkin untuk ditingkatkan. Dengan menerapkan PSO untuk *seleksi fitur* dan NBC sebagai algoritma klasifikasi, diharapkan mampu meningkatkan *accuracy* dari dokumen berita *20 Newsgroups* dengan 20 kategori.

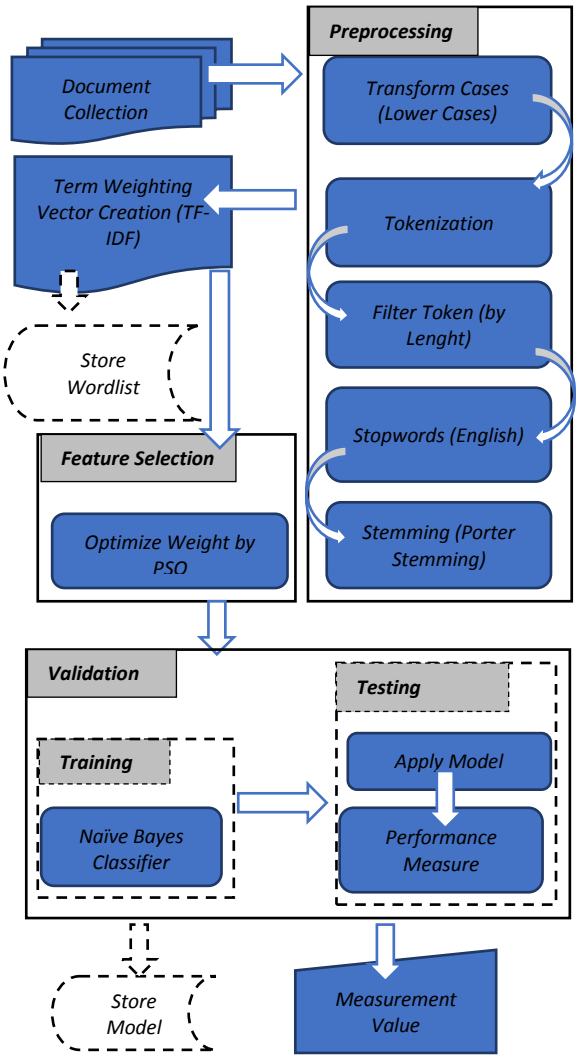
2. Metode Penelitian

Penelitian ini akan mengusulkan sebuah metode baru untuk melakukan klasifikasi terhadap teks. Metode tersebut akan diujicoba menggunakan *software* Rapidminer. Dengan menggunakan *Particle Swarm Optimization* (PSO) untuk menyeleksi fitur, dan algoritma *Naïve Bayes Classifier* (NBC) sebagai algoritma klasifikasi. Pada tahap klasifikasi akan dilakukan validasi silang (*Cross Validation*) dengan 10 kali validasi untuk mendapatkan hasil nilai *accuracy* yang maksimal. Sedangkan metode pemilihan sampel menggunakan *stratified sampling*. Hal ini dilakukan karena bentuk dari *20 Newsgroups* yang terdiri dari beberapa kategori. Sehingga data yang dikandung menjadi heterogen. Namun pada dasarnya data setiap kategori adalah homogen.

Model desain ini akan melakukan pemrosesan data *training* dan *testing* untuk menguji metode algoritma yang digunakan. Tahapan yang akan dilalui dibagi menjadi tiga bagian, yaitu *preprocessing*, seleksi fitur (*feature selection*), dan *validation* yang di dalamnya berisi sub proses *training* dan *testing* juga *performance measure* (lihat Gambar 2).

a. Pengumpulan Data

Data yang digunakan merupakan data sekunder, karena diperoleh dari *repository* peneliti yang menyediakan data tersebut, yaitu *20 newsgroups*. Berisi kumpulan *Usenet* artikel, dengan penghapusan data yang terjadi *redudant.* Masalah yang harus dipecahkan adalah, klasifikasi terhadap dokumen teks dengan akurasi yang paling maksimal. Sedangkan data yang digunakan memiliki jumlah yang cukup besar yaitu 18.846 dokumen, yang terbagi dalam 20 kategori dengan pembagian besaran dokumen seperti terlihat pada Tabel 1.



Gambar 2. Model desain penelitian yang diusulkan

Tabel 1. Jumlah dokumen setiap kategori 20 *Newsgroups datasets*

No	Nama Kategori	Jumlah
1	alt.atheism	799
2	comp.graphics	973
3	comp.os.ms-windows.misc	985
4	comp.sys.ibm.pc.hardware	982
5	comp.sys.mac.hardware	963
6	comp.windows.x	988
7	misc.forsale	975
8	rec.autos	990
9	rec.motorcycles	996
10	rec.sport.baseball	994
11	rec.sport.hockey	999
12	sci.crypt	991
13	sci.electronics	984
14	sci.med	990
15	sci.space	987
16	soc.religion.christian	997
17	talk.politics.guns	910
18	talk.politics.mideast	940

19	talk.politics.misc	775
20	talk.religion.misc	628
Total dokumen		**18.846**

Data yang sudah diperoleh, tidak langsung diproses melainkan akan dibagi lagi dengan proporsi 10% hingga 90%. Pembagian tersebut dimaksudkan untuk pengujian data *training*. Sehingga akan didapatkan hasil yang paling maksimal dari pembagian proporsi jumlah dokumen tersebut.

b. *Preprocessing*

Tahap ini akan melibatkan sub proses antara lain *Transform Case, Tokenization, Filter Token* (*by Length*), *Stopwords* (*Eglish*), dan *Stemming* (*Porter Stemming*). Hasil akhirnya berupa kumpulan kata yang sudah bersih atau unik.

c. *Vector Creation*

Setelah semua tahap *preprocessing* dilalui, maka tahap selanjutnya adalah memproses hasilnya keluarannya. Proses tersebut adalah membuat vektor kata menggunakan pembobotan TF-IDF. Pada saat yang bersamaan, hasilnya akan disimpan sebagai daftar kata (*word list*) yang akan digunakan untuk melakukan *testing* terhadap sampel dokumen yang berbeda.

d. Seleksi Fitur

Dimana akan dipilih kata yang sudah menjadi *token*, yang paling merepresentasikan dokumen. Semua kata yang sudah menjadi *token* akan dibuat menjadi vektor-vektor menggunakan perhitungan TF-IDF pada tahap *preprocessing*. Kemudian, masuk ke tahap seleksi fitur untuk dioptimasi menggunakan PSO. Partikel-partikel dalam PSO akan mencari dan menentukan *token* mana saja yang paling baik untuk dijadikan fitur. Dengan dipilihnya *token-token* sebagai fitur yang paling baik, maka akan semakin berkurang dimensi dari dokumen. Namun, isi yang dikandung oleh dokumen tetap terjaga karena fitur yang dipilih sangat merepresentasikan dokumen tersebut.

e. *Validation*

Tahap utama dari penelitian ini adalah klasifikasi, dengan menggunakan algoritma *Naïve Bayes Classifier*. Pada tahap ini, akan dilakukan perhitungan statistik, untuk mengetahui kemungkinan (probabilitas) sebuah dokumen masuk ke dalam klasifikasi (kelas) tertentu. Fitur yang sudah dipilih sebelumnya akan digunakan sebagai masukan perhitungan oleh *Naïve Bayes*, untuk mengklasifikasikan dokumen. Pada tahap ini digunakan dokumen *training* sebagai dokumen masukan.

Tahap ini digunakan untuk mengaplikasikan model yang sudah dibuat sebelumnya. Dengan menggunakan dokumen *training* sebagai dokumen *testing*, akan dilakukan perhitungan kembali untuk mengetahui tingkat kesuksesan klasifikasi pada tahap *training*. Tahap *training* dan *testing* akan divalidasi menggunakan *cross validation* dengan 10 kali validasi.

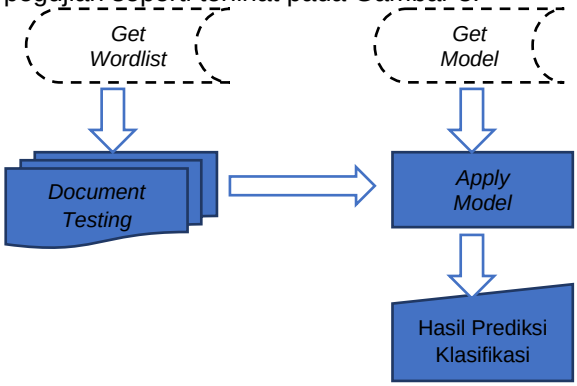
Hasil dari tahap ini adalah nilai *precision*, *recall*, dan tentunya *accuracy*. Nilai inilah yang akan dibandingkan untuk mengetahui model manakah yang paling baik. Semua hasil dari validasi akan menghasilkan model dan hasil perhitungan kinerja. Selanjutnya hasil akan ditampilkan dalam bentuk tabel *confusion matrix*, dan pada saat yang bersamaan bentuk model yang sudah dibuat akan disimpan. Bentuk model yang disimpan akan digunakan untuk melakukan *testing* terhadap sampel dokumen yang berbeda.

f. *Measurement Value*

Merupakan hasil akhir berupa tabel (confusion matrix) atau grafik yang menunjukkan nilai-nilai dari precision, recall, dan accuracy, serta time execution. Namun, pada penelitian ini time execution tidak akan dibahas. Karena sangat bergantung dari spesifikasi hardware yang digunakan.

g. Eksperimen dan Pengujian Model

Hasil dari model penelitian yang sudah didapat, akan diterapkan untuk menguji sampel dokumen yang berbeda. Akan dilakukan 2 pengujian yaitu dengan kelas dan proporsi yang sudah ditentukan, dan dengan kelas yang belum ditentukan. Model pegujian seperti terlihat pada Gambar 3.



Gambar 3. Model desain pengujian dengan dokumen sampel yang berbeda

Get wordlist merupakan *wordlist* yang sudah disimpan dari model desain sebelumnya, dan diambil dengan model

yang menggunakan proporsi dokumen *training* paling baik yaitu sebesar 80%. Kemudian untuk *get model* juga dilakukan hal yang sama. Untuk tahap *document testing*, akan digunakan dokumen dari *20 newsgroups* dengan sampel yang berbeda dari sebelumnya. Setelah melakukan proses *wordlist* kemudian melakukan tes terhadap *document testing*, maka dilanjutkan dengan perhitungan klasifikasi sesuai model yang sudah disimpan sebelumnya. Hasil akhir akan menghasilkan prediksi benar dan salah terhadap kelas yang seharusnya.

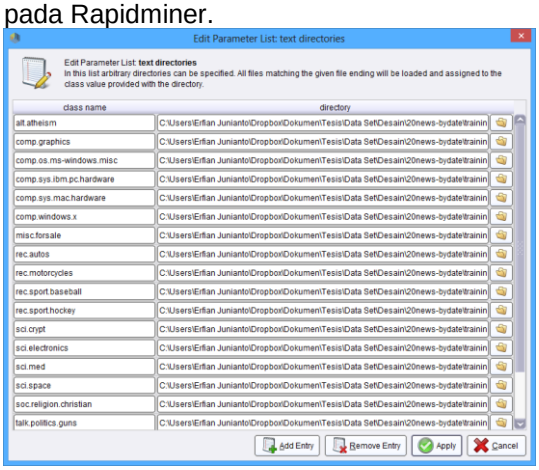
3. Hasil dan Pembahasan

3.1. Hasil Penelitian

Penelitian menghasilkan keluaran yang dapat dianalisa untuk mendapatkan informasi yang berguna. Berikut penjabaran dari hasil penelitian yang sudah dilakukan.

a. *Document Collection*

Pada Rapidminer, *document colecction* merupakan tahap awal yang digunakan sebagai masukan. Terdapat beberapa parameter, yaitu *class name* dan *directory*. *Class name* digunakan untuk memberikan nama *class* yang sudah ditentukan. Sedangkan *directory* digunakan untuk mengambil dokumen yang akan dijadikan masukan proses, dari tempat penyimpanan di dalam komputer. Berikut gambar 4 yang merupakan tampilan *document collection* pada Rapidminer.



Gambar 4. *Document Collection* pada Rapidminer

b. *Preprocessing*

Tahap yang dilalui setelah *document collection* adalah *preprocessing*. Di dalamnya terdapat tahapan-tahapan sebagai berikut.

1) *Transform Cases*

Rapidminer akan mengubah kapitalisasi karakter (huruf) menjadi kecil untuk semua kata atau huruf. Setelah melalui tahap *transform cases*, seluruh isi dokumen menjadi non kapital. Selanjutnya akan diproses pada tahap *tokenization*.

2) *Tokenization*

Proses *tokenization* dilakukan setelah *transform cases*. Semua karakter yang tidak diperlukan akan dibuang. Termasuk *white space* yang berlebihan dan semua tanda baca. Proses ini akan dilakukan terhadap setiap dokumen yang dimasukkan dari *document collection*. Sehingga diperoleh kata yang unik dan dapat merepresentasikan dokumen.

3) *Filter Token* (*by Length*)

Pada tahap ini dilakukan pemilihan *token* dengan ukuran panjang minimal 3 karakter (huruf), walaupun beberapa diantaranya adalah *stopwords*. Jika kata yang kurang dari 3 karakter (huruf), namun termasuk dalam *stopwords* akan tetap dibuang, karena tahap selanjutnya adalah pembuangan *stopwords*. Misalnya kata seperti "*dt", "vr", "i", "d", "md", "e",* dan *"s"* merupakan kata yang kurang dari 3 huruf, namun bukan merupakan *stopwords*. Hal ini tidak dapat ditangani tahap *stopwords*. Sedangkan kata seperti *"is", "a", "of", "be", "by", "in", "on", "or", "as", "to"* merupakan kata yang kurang dari 3 huruf namun merupakan *stopwords*. Jika kata tersebut terlewat pada tahap ini, masih bisa dibuang pada tahap *stopwords*.

4) *Stopwords*

Tahap *stopwords* ini akan menyempurnakan tahap *filter token by length*. Kata yang terdiri lebih dari 3 huruf dan termasuk dalam *stopwords* akan dibuang. Karena kata tersebut tidak mencerminkan isi dokumen walaupun sering muncul.

5) *Stemming*

Semua kata yang telah dipilih untuk menjadi token pada tahapan sebelumnya, akan diubah ke dalam bentuk akar (asal) kata.

Beberapa contoh kata yang diubah ke bentuk akarnya yaitu:

Sebelum Proses		Setelah Proses
oasys	→	oasi
navy	→	navi

presentations → present
expires → expir
article → articl

6) *Vector Creation*

Setelah semua tahapan *preprocessing* dilakukan, hasil keluarannya adalah daftar kata dan vektor kata. Contoh hasil yang didapatkan melalui pengolahan menggunakan Rapidminer sebagai berkut (gambar 5).

Wordlist:

Word	Total Occurences	Document Occurences	alt.atheism	comp.graphics
altern	2	1	2	0
america	1	1	1	0
american	3	1	3	0
amherst	1	1	1	0
amus	1	1	1	0
ancient	1	1	1	0
anselm	1	1	1	0
anthologi	3	1	3	0
appendix	2	1	2	0
applic	1	1	0	1

Gambar 5. Hasil proses *preprocessing* berupa *wordlist* dan frekuensi kemunculannya dalam dokumen (2 dokumen)

c. Seleksi Fitur

Seleksi fitur yang digunakan pada penelitian ini adalah PSO. Contoh hasil dari proses optimasi fitur oleh PSO menggunakan Rapidminer sebagai berikut (lihat gambar 6).

attribute	weight
franc	0.999
swinburn	0.997
defin	0.993
induct	0.992
phishnet	0.991
paraphernalia	0.991
receiv	0.990
unit	0.987
critic	0.987
repli	0.987
quotat	0.986
assort	0.985
summari	0.984
lipman	0.984

Gambar 6. Hasil proses *Optimize Weight by* PSO menggunakan Rapidmier

Ditunjukkan pada Gambar tersebut. hasil proses optimasi oleh PSO, yang mana bobot (*weight*) bisa mencapai 0,999. Hal ini tentu dapat mempengaruhi peningkatan hasil *accuracy* pada proses klasifikasi.

3.2. Evaluasi dan Validasi Hasil

Setelah proses *preprocessing* dan seleksi fitur, akan dilakukan proses klasifikasi melaui tahap *validation*. Hasil yang diperoleh dari pengujian dengan menggunakan model yang sudah diusulkan dapat dijabarkan sebagai berikut.

a. Hasil Pengujian dengan PSO-NBC

Hasil yang diperoleh pada klasifikasi yang menggunakan proporsi dokumen *training* dari 10% hingga 90% adalah:

Tabel 2. Hasil *precision*, *recall*, *accuracy* pengujian menggunakan PSO-NBC

No	Algoritma	Proporsi Dokumen	Jumlah Dokumen	Hasil		
				Precision	*Recall*	*Accuracy*
1	PSO-NBC	10%	1885	78,32%	78,12%	78,21%
2	PSO-NBC	20%	3769	79,59%	80,07%	79,92%
3	PSO-NBC	30%	5654	81,52%	81,75%	81,72%
4	PSO-NBC	40%	7538	82,98%	83,25%	83,25%
5	PSO-NBC	50%	9423	83,64%	83,87%	83,86%
6	PSO-NBC	60%	11308	84,19%	84,34%	84,37%
7	PSO-NBC	70%	13192	84,54%	84,70%	84,74%
8	PSO-NBC	80%	15077	**85,21%**	**85,40%**	**85,42%**
9	PSO-NBC	90%	16961	85,11%	85,27%	85,29%

Dari Tabel 2 tersebut dapat diketahui bahwa proporsi dokumen *training* yang menghasilkan *accuracy* paling tinggi adalah 80%. Dengan hasil *accuracy* sebesar 85,42%. Dapat dilihat bahwa semakin tinggi dokumen *training*, maka akan semakin tinggi pula akurasinya.

b. Hasil Testing dengan Sampel Dokumen yang Berbeda

1) Kelas sudah ditentukan

Dilakukan pengujian dengan model yang sudah disimpan dan dipilih model terbaik (80%), serta menggunakan dokumen *sample* yang berbeda. Kemudian digunakan proporsi dokumen *testing* dari 10% hingga 90%. Namun, kelas dokumen sampel sudah ditentukan. Berikut hasil pengujiannya.

Tabel 3. Hasil pengujian dengan dokumen sampel berbeda dan kelas yang sudah ditentukan

No	Proporsi		Hasil Testing			
	Model	Testing	Pred. Benar	Pred. Salah	Total	*Accuracy*
1	80%	90%	16486	477	16963	97,19
2	80%	80%	15055	21	15076	99,86
3	80%	70%	13174	20	13194	99,85
4	80%	60%	11287	20	11307	99,82
5	80%	50%	9410	18	9428	99,81
6	80%	40%	7525	14	7539	99,81
7	80%	30%	5646	9	5655	99,84
8	80%	20%	3765	5	3770	99,87
9	80%	10%	1883	3	1886	99,84

Dari Tabel 3 tersebut diketahui bahwa *accuracy* paling tinggi didapat pada pengujian dengan proporsi dokumen *testing* sebesar 20% dengan nilai akurasi 99,87%. Terbukti bahwa model

klasifikasi dengan menggunakan PSO untuk seleksi fitur dapat meningkatkan akurasi, daripada penelitian sebelumnya yang hanya menggunakan NBC. Dengan *datasets* yang sama yaitu *20 Newsgroups* (Isa, Hong, Kallimani, & Rajkumar, 2008).

2) Kelas belum ditentukan

Dilakukan pengujian dengan model yang sudah disimpan dan dipilih model terbaik (80%), serta menggunakan dokumen sampel yang berbeda. Namun, kelas dokumen sampel belum ditentukan. Pada pengujian yang tersaji dalam Tabel 4 dipilih satu dokumen dari masing-masing kategori untuk mewakili data sampel.

Tabel 4. Hasil Pengujian dengan dokumen sampel berbeda dan kelas yang belum ditentukan

No	Dokumen	Kelas Sebenarnya	Kelas Prediksi	Hasil
1	54175	alt.atheism	alt.atheism	Benar
2	39063	comp.graphics	misc.forsale	Salah
3	10797	comp.os.windows.misc	misc.forsale	Salah
4	61081	comp.sys.ibm.pc.hardware	comp.sys.ibm.pc.hardware	Benar
5	52255	comp.sys.mac.hardware	comp.graphics	Salah
6	68198	comp.windows.x	comp.windows.x	Benar
7	76795	misc.forsale	misc.forsale	Benar
8	103702	rec.autos	rec.autos	Benar
9	105136	rec.motorcycles	rec.motorcycles	Benar
10	105057	rec.sport.baseball	rec.sport.baseball	Benar
11	54541	rec.sport.hockey	rec.sport.hockey	Benar
12	16046	sci.crypt	sci.crypt	Benar
13	54285	sci.electronics	sci.space	Salah
14	59541	sci.med	sci.med	Benar
15	61552	sci.space	sci.space	Benar
16	21709	soc.religion.christian	soc.religion.christian	Benar
17	55094	talk.politics.guns	talk.politics.guns	Benar
18	77330	talk.politics.mideast	talk.politics.mideast	Benar
19	179024	talk.politics.misc	talk.politics.misc	Benar
20	84351	talk.religion.misc	talk.politics.misc	Salah

Diketahui bahwa sebanyak 20 dokumen yang diuji, terdapat 15 dokumen dengan prediksi benar dan 5 dokumen prediksi salah. Sehingga jika dihitung nilai *accuracy* akan didapatkan hasil sebesar 75%. Ini membuktikan bahwa pengujian dengan dokumen yang sangat sedikit, masih memiliki nilai *accuracy* yang tinggi.

4. Kesimpulan

Dari hasil evaluasi dan validasi diketahui bahwa algoritma PSO dapat diterapkan sebagai seleksi fitur dalam klasifikasi teks, dengan bentuk data yang tidak terstruktur. Algoritma PSO yang diterapkan sebagai seleksi fitur, mampu mengurangi dimensi dokumen yang sangat besar sekaligus meningkatkan *accuracy*.

Algortima NBC juga mampu menangani klasifikasi dengan jumlah data yang besar, dan kategori yang cukup banyak. Hasil akurasi dari dokumen *training* yang didapat adalah 85,42%, dalam pengujian dengan proporsi dokumen *training* sebesar 80% (15077 dokumen). Sedangkan pengujian menggunakan dokumen *testing* menghasilkan akurasi sebesar 99,87% dengan proporsi dokumen *testing* sebesar 20% (3370 dokumen).

Diketahui bahwa dokumen teks memliki struktur yang tidak teratur. Maka tahap *preprocessing* sangat dibutuhkan untuk pemrosesan data awal. Karena terbukti dapat mengurangi dimensi dokumen dan membantu membuang kata yang tidak perlu. Sehingga kinerja tahap seleksi fitur dan klasifikasi menjadi lebih mudah dengan hasil yang lebih akurat.

Meskipun PSO dapat diterapkan sebagai seleksi fitur, dan mampu meningkatkan akurasi, masih perlu penelitian lebih lanjut. Misalnya, dalam penelitian ini belum dilakukan optimasi dengan menggunakan parameter *inertia weight*. Sehingga PSO masih memiliki kemungkinan untuk lebih meningkatkan *accuracy*. Untuk penerapan lebih lanjut, algoritma NBC dan PSO dapat digunakan sebagai moderator pendeteksi kelayakan *posting* pada sebuah forum atau berita *online*.

Referensi

Andri. (2014, Mei 07). *Jurnalisme Digital: Crowdsourcing Berita Jurnalis*. Dipetik 05 14, 2014, dari Institut Komunikasi Indonesia Baru: http://komunikasi.us/index.php/course/17-pengantar-teknologi-informasi-dan-komunikasi/1479-jurnalisme-digital-crowdsourcing-berita-jurnalis

Dalal, M. K., & Zaveri, M. A. (2011). Automatic Text Classification: A Techical Review. *International Journal of Computer Applications, 28*(2), 37-40.

Destuardi, I., & Sumpeno, S. (2009). Klasifikasi Emosi untuk Teks Berbahasa Indonesia Menggunakan Metode Naive Bayes. *Seminar Nasional Pascasarjana IX-ITS*.

Dewi, I. N., & Supriyanto, C. (2013). Klasifikasi Teks Pesan Spam Menggunakan Algoritma Naive

Bayes. *Seminar Nasional Teknologi Informasi & Komunikasi Terapan*, 156-160.

Feldman, R., & Sanger, J. (2007). *The Text Mining Handbook Advanced Approaches in Analyzing Unstructured Data.* Cambridge: Cambridge University Press.

Ghosh, S., Roy, S., & Bandyopadhyay, S. K. (2012). A Tutorial Review on Text Mining Algorithms. *International Journal of Advanced Research in Computer and Communication Engineering*, 223-233.

Hamzah, A. (2012). Klasifikasi Teks dengan Naive Bayes Classifier (NBC) untuk Pengelompokan Teks Berita dan Abstrak Akademis. *Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST) Periode III, Yogyakarta*, 269-277.

Isa, D., Hong, L. L., Kallimani, V., & Rajkumar, R. (2008). Text Document Pre-Processing using Bayes Formula for Classification Based on the Vector Space Model. *Computer and Information Science, 1*(4), 79-90. Retrieved 2014

Korde, V., & Mahender, C. (2012). Text Classification and Classifier: A Survey. *International journal of Artificial Intelligence & Applications (IJAIA)*, 85-99.

Polettini, N. (2004). The Vector Space Model in Information Retrieval - Term Weighting Problem. 1-9.

Samodra, J., Sumpeno, S., & Hariadi, M. (2009). Klasifikasi Dokumen Teks Berbahasa Indonesia dengan Menggunakan Naive Bayes. *Seminar Nasional Electrical, informatic, and it's Education*.

Syafrullah, M., & Salim, N. (2010). Improving Term Extraction Using Particle Swarm Optimization Techniques. *Journal of Computing, 2*(2), 116-120.

Tu, C.-J., Chuang, L.-Y., Chang, J.-Y., & Yang, C. (2007). Feature Selection Using PSO-SVM. *IAENG Intarnational Journal of Computer Science*.

Wahono, R. S., & Suryana, N. (2013). Combining Particle Swarm Optimization based Feature Selection and Bagging Technique for Software Defect Prediction. *International Journal of Software Engineering and Its Applications, 7*(5), 153-166.

Weiss, S. M., Indurkhya, N., Zhang, T., & Damerau, F. J. (2005). *Text Mining: Predictive Methods for Analyzing Unstructured Information.* United States of America: Springer.

Wibisono, Y. (2005). Klasifikasi Berita Berbahasa Indonesia menggunakan Naive Bayes. *Seminar Nasional Matematika Universitas Pendidikan Indonesia*.

Widiasri, M., Justitia, A., & Arifin, A. Z. (2011). Penerapan Particle Swarm Optimization untuk Penentuan Parameter Regularisasi pada Kernel Regularized Discriminant Analysis. *Industrial Electronics Seminar*, 61-66.

Xue, B., Zhang, M., & Browne, W. (2012). Multi-Objective Particle Swarm Optimization (PSO) for Feature Selection. *GECCO'12*.