Online News Classification Using Naive Bayes Classifier with Mutual Information for Feature Selection

Shafrian Adhi Karunia
Program Studi Informatika
Fakultas MIPA
Universitas Sebelas Maret
shafrian@student.uns.ac.id

Ristu Saptono
Program Studi Informatika
Fakultas MIPA
Universitas Sebelas Maret
ristu.saptono@staff.uns.ac.id

Rini Anggrainingsih
Program Studi Informatika
Fakultas MIPA
Universitas Sebelas Maret
rini.anggrainingsih@staff.uns.ac.id

ABSTRACT

The number of online news documents can reach billion documents. Therefore, the grouping of news documents required to facilitate a editorial staff to input and categorize news by its categories.

This paper aim to classify online news using Naive Bayes Classifier with Mutual Information for feature selection that aims to determine the accuracy from combination of this methods in the classification of online news documents, so grouping of online news documents can be classified automatically and achieve more accurate for classification model. Data is divided into training and testing data. Data in August, September and October 2016 was used for training data. For testing data, 65 documents was used that located in November. The best results of this methods are 80% of accuracy, 94.28% of precision, 79.68% of recall and 85.08% of f-measure for Multivariate Bernoulli without feature selection. Then the best results of classification model using Mutual Information for feature selection achieved in Multivariate Bernoulli model with 70% of accuracy, 89.11% of precision, 69.76% of recall and 78.04% of f-measure with the word's efficiency rate until 52% than before using feature selection. In other hand, the results of Multinomial Naïve Bayes without feature selection are 41.67% of accuracy, 75.68% of precision, 41.90% of recall and 48.13% of f-measure, for the results of Multinomial Naïve Bayes model using feature selection are 10% of accuracy, 33.33% of precision, 9.40% of recall and 14.35% of f-measure.

Keywords

Classification, Feature Selection, Multinomial Naive Bayes, Multivariate Bernoulli, Mutual Information, Naïve Bayes Classifier

1 PENDAHULUAN

Jumlah dokumen berita yang dikonversi ke dalam media internet selalu bertambah bahkan dapat mencapai milyaran dokumen berita. Oleh karenanya, pengelompokan dokumen berita dibutuhkan untuk mempermudah dalam pencarian informasi berita online mengenai topic tertentu [1] serta memudahkan penulis berita dalam melakukan input berita dan mengkategorikannya sesuai kategori yang ada. Dari paparan diatas, sebuah solusi diusulkan yaitu otomatisasi klasifikasi berita online berdasarkan kategori yang sudah ada.

Terdapat beberapa pendekatan untuk melakukan klasifikasi teks antara lain pendekatan *Probabilistic*, *Support Vector Machine* dan *Artificial Neural Network* atau *Decision Tree Classification*. Metode Naïve bayes adalah metode *probabilistic* yang merupakan metode klasifikasi yang secara umum sering digunakan karena kesederhanaan dalam komputasinya [2]. Dalam penelitian yang telah dilakukan, metode Naïve Bayes dapat digunakan secara efektif untuk mengklasifikasikan dokumen teks berbahasa Indonesia dengan akurasi mencapai 87,63% [3]. *Naïve Bayes classifier* dapat dibagi menjadi dua, yaitu *Multivariate Bernoulli* dan *Multinomial Naïve Bayes* [4]. *Multinomial Naïve Bayes* mampu menurunkan kesalahan pada klasifikasi dokumen dengan nilai rata-rata 27% bahkan mencapai 50% dari percobaan menggunakan *Multivariate Bernoulli* [5]. Penelitian mengenai perbandingan kinerja dua model *Naïve Bayes Classifier* pada kasus *Anti-Spam Email Filtering* menghasilkan bahwa model *Multinomial Naïve Bayes* dapat mencapai akurasi yang lebih baik dibandingkan dengan *Multivariate Bernoulli* dengan seleksi fitur *Mutual Information* [6]. Berdasarkan penelitian terkait tersebut, penelitian ini menggunakan kedua metode *Naïve Bayes Classifier* yang diharapkan dapat memperoleh model klasifikasi berita online dengan hasil akurasi yang maksimal.

Dalam rangka meningkatkan tingkat akurasi klasifikasi dan efisiensi fitur maka perlu melakukan seleksi fitur. Dari penelitian yang telah dilakukan, yang membandingkan tiga seleksi fitur yaitu *Invers Document Frequency*, *Mutual Information* dan *Chi-square*, pada *spam filter* menggunakan klasifikasi *Multinomial Naïve Bayes* didapatkan hasil bahwa seleksi fitur dengan *Mutual Information* memiliki nilai akurasi tertinggi dengan angka 93.77% dibandingkan dua metode yang lain [7]. Berdasarkan penelitian tersebut, dalam penelitian ini diusulkan perbandingan metode *Multivariate Bernoulli* dan metode *Multinomial Naïve Bayes* untuk pengklasifikasian berita online dengan penambahan seleksi fitur *Mutual Information* yang diharapkan dapat memberikan hasil yang lebih maksimal.

2 TEXT PREPROCESSING

Text Preprocessing adalah proses pengubahan bentuk data yang tidak terstruktur menjadi data yang terstruktur sesuai kebutuhan untuk proses dalam text mining. Tahap *preprocessing* terdiri dari *case folding*, *tokenizing*, *filtering*, *stemming*, *tagging*, dan *analyzing* [8]. Dalam penelitian ini digunakan tahap *case folding* hingga *filtering*. Pertama, *Case folding* adalah tahap mengubah semua huruf dalam dokumen menjadi huruf kecil. Selain itu, karakter non-huruf akan dihilangkan [8]. Selanjutnya *Tokenizing* adalah tahap pemecahan kalimat berdasarkan tiap kata yang menyusunnya [8]. Terakhir *Filtering* adalah tahap mengambil kata-kata penting dari hasil tahap *tokenizing*. *Filtering* dapat dilakukan dengan menghilangkan *stoplist/stopword* (kata-kata yang tidak deskriptif, seperti kata "yang" dan "dari") [8] [9].

3 DATA PREPROCESSING

Ada beberapa teknik atau metode dalam melakukan *Data Preprocessing*, antara lain *data cleaning, data integration, data reduction* dan *data transformation. Data cleaning* dapat diterapkan untuk menghilangkan noise dan memperbaiki inkonsistensi data. *Data integration* menggabungkan data dari beberapa sumber sehingga menjadi data yang koheren seperti sebuah *data warehouse*. *Data reduction* dapat mereduksi ukuran data dengan mengeliminasi fitur yang redundan atau *clustering*. *Data transformations* dapat diterapkan dengan mengubah skala ke dalam range yang lebih kecil seperti 0.0 hingga 1.0. Langkah-langkah ini dapat meningkatkan akurasi dan efisiensi daripada algoritma *data mining* itu sendiri. Teknik ini tidak harus selalu berdiri sendiri, tetapi bisa digabungkan antara satu dengan yang lain.

Dalam penelitian ini digunakan metode *data cleaning* untuk *Data Preprocessingnya*. *Data cleaning* merupakan salah satu metode yang digunakan untuk *data preprocessing* dalam *data mining.* Metode ini dilakukan dengan menghilangkan nilai-nilai data yang salah, memperbaiki kekacauan data dan memperbaiki data yang tidak konsisten [10].

4 MUTUAL INFORMATION

Salah satu seleksi fitur yang sering digunakan untuk menghitung bobot dari fitur adalah *Mutual information*. MI menunjukan seberapa banyak informasi ada atau tidaknya sebuah term memberikan konstribusi dalam membuat keputusan klasifikasi secara benar. Nilai dari mutual information disimbolkan dengan notasi MI, seperti pada persamaan (1).

$$MI(F,C) = \sum_{et\varepsilon\{1,0\}} \sum_{ec\varepsilon\{1,0\}} P(F=et, C=ec) \ln \left(\frac{P(F=et,C=ec)}{P(F=et)P(C=ec)}\right) \quad (1)$$

dengan F adalah variabel acak yang mengambil dengan nilai-nilai et = 1 (dokumen berisi term t) dan et = 0 (dokumen tidak mengandung t), dan C adalah variabel acak yang mengambil dengan nilai-nilai ec = 1 (dokumen di kelas c) dan ec = 0 (dokumen tidak di kelas c). Nilai dari MI juga bisa dijabarkan menjadi persamaan (2).

$$MI(F,C) = \frac{NF,C}{N}\ln\frac{N.NF,C}{NF.NC} + \frac{NF,\overline{C}}{N}\ln\frac{N.NF,\overline{C}}{NF.N\overline{C}} + \frac{N\overline{F},C}{N}\ln\frac{N.N\overline{F},C}{N\overline{F}.NC} + \frac{N\overline{F},\overline{C}}{N}\ln\frac{N.N\overline{F},\overline{C}}{N\overline{F}.N\overline{C}} \quad (2)$$

dengan N adalah jumlah dokumen yang memiliki nilai-nilai et dan ec yang ditunjukan oleh dua subscript. Sebagai contoh, NF,$\overline{C}$ adalah jumlah dokumen yang mengandung t (et = 1) dan tidak dalam c (ec = 0). N adalah jumlah total dokumen atau N= NF,C + NF, $\overline{C}$ + N$\overline{F}$, C + N$\overline{F}$, $\overline{C}$ [7].

5 MULTINOMIAL NAÏVE BAYES

Multinomial naïve bayes dipengaruhi oleh serangkaian term, dengan kata lain jumlah term diperhitungkan. Peluang antara term satu dengan yang lain adalah independen (tidak bergantung). Pada Naïve Bayes, nilai peluang dokumen d dalam kelas c dapat dihitung seperti persamaan (3) [7] [9].

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)} \quad (3)$$

dengan
c adalah kelas c
d adalah dokumen
P adalah peluang

Dengan memperhitungkan term nilai peluang pada dokumen d dalam kelas c dapat dihitung dengan persamaan (4) [11].

$$P(c|d) = P(c)\prod_{1\le k\le nd} P(fk|c) \quad (4)$$

dengan
P(c|d) adalah peluang kategori c pada dokumen d
P(c) adalah nilai peluang dari kategori c
P(fk|c) adalah peluang dari suatu term fk yang ada pada kelas c

Parameter P(fk|c) dihitung dengan persamaan (5).

$$P(fk|c) = \frac{Tct+1}{\sum_{t'\epsilon VTct'} + B'} \quad (5)$$

dengan

Tct adalah kejadian term t pada sebuah dokumen dari kelas c
∑t'ϵVTct' adalah jumlah term yang terdapat pada seluruh dokumen di kelas c
B adalah jumlah variasi kata (*vocabulary*) yang terdapat dalam data latih [11].

6 MULTIVARIATE BERNOULLI

Pada model *Bernoulli*, setiap data uji dihitung nilai peluang tiap kata terhadap masing-masing kelas. Perbedaannya dengan model yang pertama yaitu perhitungan peluang menggunakan jumlah data yang mengandung kata t (Tct), bukan frekuensi kemunculan kata t [4]. Selain itu, jika model *Multinomial* menggunakan jumlah kata dan jumlah *vocabulary*, *model Bernoulli* menggunakan jumlah data latih tiap kelas (Tc) dan jumlah kelas (Σc). Peluang tiap kata dihitung menggunakan persamaan:

$$P(fk|c) = \frac{Tct+1}{Tc+\sum c} \quad (6)$$

Setelah peluang tiap kata didapatkan, nilai dari hasil perkalian invers untuk tiap peluang kata pada data latih selain kata pada data uji dihitung. Misalnya ada sekumpulan kata pada data latih d dan sekumpulan kata pada data uji f. Selanjutnya dicari kata-kata pada data latih d yang independen dengan kata-kata pada data uji f. Dari kata-kata tersebut kemudian diambil nilai peluang kata (dalam hal ini diberi simbol P(fk'|c)) dan jumlah kata (M). Nilai peluang kalimat terhadap suatu kelas dihitung dengan persamaan:

$$P(c|d) = P(c)\prod_{i=1}^{N} P(fk|c)\ x \prod_{i=1}^{M}(1 - P(fk'|c)) \quad (7)$$

7 PENGUJIAN

Pengujian dilakukan untuk menilai kecukupan model untuk dijadikan metode untuk klasifikasi berita online. Pengujian performa dari metode yang diusulkan penelitian ini dengan menggunakan *confusion matrix* yang terdiri atas empat metode, yaitu *accuracy*, *precision*, *recall* dan *f-measure* seperti ditunjukkan Tabel 1.

Tabel 1 Pengujian [12]

Kelas	Hasil Klasifikasi				Total
	C1	C2	C3	Cn	
C1	TP_C1	*Error*	*Error*	*Error*	Total_C1
C2	*Error*	TP_C2	*Error*	*Error*	Total_C2
C3	*Error*	*Error*	TP_C3	*Error*	Total_C3
Cn	*Error*	*Error*	*Error*	TP_Cn	Total_Cn
	T_C1	T_C2	T_C3	T_Cn	

$$accuracy = \frac{TP(C1)+TP(C2)+TP(C3)+TP(Cn)}{Total(C1)+Total(C2)+Total(C3)+Total(Cn)} \quad (6)$$

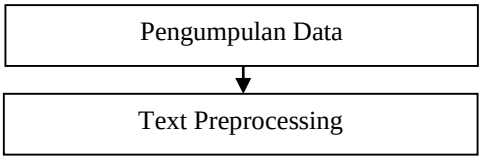
$$p(Ci) = \frac{TP(Ci)}{T(Ci)} \quad (7)$$

$$r(Ci) = \frac{TP(Ci)}{Total(Ci)} \quad (8)$$

$$F(Ci) = \frac{2p(Ci)*r(Ci)}{p(Ci)+r(Ci)} \quad (9)$$

8 METODOLOGI

Tahapan yang dilakukan dalam penelitian disajikan pada Gambar 1.



Gambar 1 Metodologi Penelitian

8.1 Pengumpulan Data

Penelitian ini menggunakan data berita online yang berupa ringkasan dari keseluruhan isi berita tersebut. Dokumen tersebut diambil dari salah satu portal web berita online Indonesia yaitu detik.com. Untuk memperoleh dokumen tersebut, digunakan sebuah *library* yang bernama *SimplePie*. *SimplePie* akan mengambil dokumen dari repositori yang telah ditentukan secara otomatis lalu disimpan di dalam database. Data yang terkumpul adalah sebanyak 2386 data, yang terdiri dari 351 pada kategori food, 341 dokumen pada kategori health, 323 dokumen pada kategori inet, 347 dokumen pada kategori otomotif, 361 dokumen pada kategori sport, 341 dokumen pada kategori travel dan 322 dokumen pada kategori wolipop. Dari jumlah data tersebut, diambil semua data training dari tiap-tiap kategori yang terdapat pada bulan Agustus, September dan Oktober untuk data training kemudian data testing sebanyak 65 dokumen yang berada pada bulan November.

8.2 Text Preprocessing

Tahap *preprocessing* terdiri dari *case folding*, *tokenizing*, *filtering*, *stemming*, *tagging*, dan *analyzing* [9]. Dalam penelitian ini digunakan tahap *case folding* hingga *filtering*. Secara umum sebuah dokumen berita online terdiri atas teks, tanda baca, angka dan lain lain. Bagian dokumen berita online yang tidak berupa teks harus dihilangkan karena dapat menurunkan kinerja klasifikasi. Tiga tahapan *text preprocessing* tersebut tidak lain adalah bertujuan menghilangkan bagian dokumen berita online yang tidak berupa teks. Dibawah ini adalah contoh dari hasil *text preprocessing* ditunjukkan Gambar 2.

Setelah dilakukan *text preprocessing*, teknik *data preprocessing* dilakukan dengan menggunakan *data cleaning*. Teknik ini bertujuan untuk menghilangkan nilai-nilai data yang salah, memperbaiki kekacauan data dan memperbaiki data yang tidak konsisten. Dalam implementasinya teknik ini dilakukan dengan menghapus data dengan frekuensi terkecil berdasarkan jumlah kata yang ada pada masing-masing berita. Teknik tersebut dilakukan pada semua kategori baik *data training* maupun *data testing*.

Teks awal	Teks hasil *Text Preprocessing*
langkah kapolri tangani provokasi di medsos lacak isu dan gandeng menkominfo untuk mencegah isu menjadi liar dan berujung provokasi kerjasama dengan kemenkominfo menjadi langkah strategis	langkah menkominfo langkah kapolri mencegah strategis tangani isu provokasi liar medsos berujung lacak provokasi isu kerjasama gandeng kemenkominfo

Gambar 2 Contoh hasil *Text Preprocessing*

8.3 Penerapan Seleksi Fitur Mutual Information

Pada tahap ini dilakukan penerapan seleksi fitur dengan metode *Mutual Information*. Tahap ini bertujuan agar akurasi yang dihasilkan dari proses klasifikasi berita online dapat meningkat, serta yang terpenting adalah untuk efisiensi dari fitur yang digunakan sebagai penciri. Dalam implementasinya, seleksi fitur diberikan *threshold* yang diharapkan dapat menyeleksi fitur secara lebih efektif sehingga hasil klasifikasi lebih akurat. Nilai dari *threshold* itu sendiri tidak ada batasan khusus untuk menentukannya tetapi menggunakan batasan sesuai yang kita inginkan [7].

8.4 Klasifikasi

Pada tahap ini dilakukan penerapan metode klasifikasi *Multinomial Naïve Bayes* dan *Multivariate Bernoulli*. Metode *Multinomial Naïve bayes* dan *Multivariate Bernoulli* dalam penelitian ini diterapkan untuk mengklasifikasikan berita yang ada ke dalam kelas yang telah ditentukan.

8.5 Pengembangan Aplikasi

Pada tahap ini dilakukan pengembangan aplikasi menggunakan bahasa pemrograman PHP dengan *database* MySQL.

8.6 Pengujian

Pengujian dilakukan untuk menilai kecukupan model untuk dijadikan metode untuk klasifikasi berita online. Pengujian performa dari metode yang diusulkan penelitian ini dengan menggunakan *confusion matrix* yang terdiri atas empat metode, yaitu *accuracy, precision, recall* dan *f-measure.*

9 HASIL DAN PEMBAHASAN

Dalam klasifikasi *Multinomial Naive Bayes* dan *Multivariate Bernoulli* digunakan data training berjumlah 2321, yaitu 341 dokumen pada kategori food, 333 dokumen pada kategori health, 337 dokumen pada kategori otomotif, 352 dokumen pada kategori sport, 314 dokumen pada kategori inet, 331 dokumen pada kategori travel dan 313 dokumen pada kategori wolipop. Sedangkan data training yang digunakan setelah dilakukan *data cleaning* berjumlah 2308, yaitu 339 dokumen pada kategori *food*, 331 dokumen pada kategori *health*, 337 dokumen pada kategori *otomotif*, 350 dokumen pada kategori *sport*, 311 dokumen pada kategori *inet*, 329 dokumen pada kategori *travel* dan 311 dokumen pada kategori *wolipop*. Hasil dari pembuangan *stopwords* sebelum dilakukan proses seleksi fitur menghasilkan jumlah token unik atau *vocabulary* sebanyak 7785 kata.

Dalam implementasinya, seleksi fitur diberikan *threshold* yang diharapkan dapat menyeleksi fitur secara lebih efektif sehingga hasil klasifikasi lebih akurat. Nilai dari *threshold Mutual Information* itu sendiri tidak ada batasan khusus untuk menentukannya tetapi menggunakan batasan sesuai yang kita inginkan. Nilai awal dari *threshold* seleksi fitur dalam peneletian ini diperoleh dari penelitian terkait yang ada yaitu 0.0004. Dari *threshold* tesebut kemudian dilakukan perubahan dengan memperkecil nilai *threshold* seperti terlihat pada Tabel 5 dan Tabel 7. Hasil dari seleksi fitur pada masing-masing kategori dapat dilihat pada Tabel 2 dan Tabel 3.

Tabel 2 Hasil Seleksi Fitur Sebelum *Data* Cleaning

Threshold	Jumlah Kata per Kelas						
	Food	*Health*	*Otomotif*	*Sport*	*Inet*	*Travel*	*Wolipop*
-	7785	7785	7785	7785	7785	7785	7785
0.0004	1602	1562	1531	1686	1639	1588	1472

Tabel 2 Hasil Seleksi Fitur Sebelum *Data* Cleaning

Threshold	Jumlah Kata per Kelas						
	Food	*Health*	*Otomotif*	*Sport*	*Inet*	*Travel*	*Wolipop*
0.0002	2708	2699	2633	2787	2391	2347	2238
0.00012	3863	3819	3792	3879	3907	3917	3828

Tabel 3 Hasil Seleksi Fitur Setelah *Data* Cleaning

Threshold	Jumlah Kata per Kelas						
	Food	*Health*	*Otomotif*	*Sport*	*Inet*	*Travel*	*Wolipop*
-	7776	7776	7776	7776	7776	7776	7776
0.0004	1599	1553	1525	1684	1632	1692	1464
0.0002	2703	2690	2634	2780	2389	2807	2234
0.00012	3855	3811	3786	3874	3897	3910	3821

Untuk menguji ingatan sistem terhadap data-data yang ada maka dilakukan *memorizing* untuk masing-masing metode. *Memorizing* disebut juga proses testing untuk data training. Semua data training yang ada dilakukan testing dengan menggunakan data training itu sendiri. Setelah semua dokumen telah selesai diklasifikasikan, dilakukan pengujian dengan menggunakan *confusion matrix* yang terdiri dari empat metode, yaitu *accuracy*, *precision*, *recall* dan *f-measure*. Perhitungan keempat metode ini dilakukan sebanyak dua kali untuk yang menggunakan teknik data cleaning dan tanpa menggunakan teknik data cleaning dengan masing-masing pengujian dilakukan empat kali pengujian untuk Multinomial Naïve Bayes dan empat kali pengujian untuk Multivariate Bernoulli, dengan percobaan pertama dari masing-masing metode klasifikasi tersebut tanpa menggunakan seleksi fitur Mutual Information, percobaan selanjutnya menggunakan fitur seleksi Mutual Information dengan threshold yang berbeda. Hasil diperoleh seperti pada Tabel 4, Tabel 5, Tabel 6 dan Tabel 7.

Tabel 4 Hasil *Memorizing Multinomial Naïve Bayes* Sebelum *Data Cleaning*

Percobaan ke-	*Accuracy*	*Precision*	*Recall*	*F-Measure*	*Threshold*
1	46.83%	75.10%	46.76%	54.41%	-
2	8.31%	40.94%	8.42%	13%	0.0004
3	10.47%	44.11%	10.52%	16.05%	0.0002
4	13.18%	46.19%	13.16%	19.23%	0.00012

Tabel 5 Hasil *Memorizing Multinomial Naïve Bayes* Setelah *Data Cleaning*

Percobaan ke-	*Accuracy*	*Precision*	*Recall*	*F-Measure*	*Threshold*
1	46.40%	75.10%	46.32%	53.98%	-
2	8.19%	40.77%	8.29%	12.82%	0.0004
3	10.40%	44.20%	10.45%	15.93%	0.0002
4	13.08%	46.37%	13.06%	19.16%	0.00012

Tabel 6 Hasil *Memorizing Multivariate Bernoulli* Sebelum *Data Cleaning*

Percobaan ke-	*Accuracy*	*Precision*	*Recall*	*F-Measure*	*Threshold*
1	93.97%	97.15%	93.88%	95.41%	-
2	40.89%	63.69%	41.72%	42.60%	0.0004

Tabel 6 Hasil *Memorizing Multivariate Bernoulli* Sebelum *Data Cleaning*

Percobaan ke-	*Accuracy*	*Precision*	*Recall*	*F-Measure*	*Threshold*
3	52.05%	73.85%	52.56%	58.43%	0.0002
4	65.23%	82.36%	65.55%	71.70%	0.00012

Tabel 7 Hasil *Memorizing Multivariate Bernoulli* Setelah *Data Cleaning*

Percobaan ke-	*Accuracy*	*Precision*	*Recall*	*F-Measure*	*Threshold*
1	94.02%	97.16%	93.94%	95.45%	-
2	40.60%	63.27%	41.49%	42.31%	0.0004
3	51.99%	73.73%	52.54%	58.24%	0.0002
4	65.42%	82.44%	65.76%	71.86%	0.00012

Selanjutnya, model yang diperoleh digunakan untuk mengklasifikasikan 65 buah data testing untuk sebelum dilakukan *data cleaning* dan 60 buah data testing untuk yang sesudah dilakukan *data cleaning*. Hasil ditunjukkan pada Tabel 8, Tabel 9, Tabel 10 dan Tabel 11.

Tabel 8 Hasil Pengujian *Multinomial Naïve Bayes* Sebelum *Data Cleaning*

Percobaan ke-	*Accuracy*	*Precision*	*Recall*	*F-Measure*	*Threshold*	Waktu Komputasi
1	40%	71.60%	39.64%	46.36%	-	35.10 s
2	9.23%	30.24%	9.44%	12.36%	0.0004	11.91 s
3	9.23%	36.31%	9.05%	13.48%	0.0002	15.52 s
4	7.69%	26.67%	7.62%	11.30%	0.00012	21.70 s

Tabel 9 Hasil Pengujian *Multinomial Naïve Bayes* Setelah *Data Cleaning*

Percobaan ke-	*Accuracy*	*Precision*	*Recall*	*F-Measure*	*Threshold*	Waktu Komputasi
1	41.67%	75.68%	41.90%	48.13%	-	34.01 s
2	5%	25%	4.80%	7.65%	0.0004	10.19 s
3	8.33%	35.37%	7.98%	12.22%	0.0002	15.00 s
4	10%	33.33%	9.40%	14.35%	0.00012	20.32 s

Tabel 10 Hasil Pengujian *Multivariate Bernoulli* Sebelum *Data Cleaning*

Percobaan ke-	*Accuracy*	*Precision*	*Recall*	*F-Measure*	*Threshold*	Waktu Komputasi
1	76.92%	93.17%	76.39%	82.80%	-	16.77 s
2	41.54%	65.24%	42.02%	46.29%	0.0004	7.49 s
3	55.38%	79.01%	55.24%	64.25%	0.0002	7.30 s
4	69.23%	89.03%	69.44%	77.49%	0.00012	9.53 s

Tabel 11 Hasil Pengujian *Multivariate Bernoulli* Sesudah *Data Cleaning*

Percobaan ke-	*Accuracy*	*Precision*	*Recall*	*F-Measure*	*Threshold*	Waktu Komputasi
1	80%	94.28%	79.68%	85.08%	-	9.81 s
2	43.33%	70%	42.78%	47.77%	0.0004	6.26 s
3	55%	77.58%	55.16%	63.36%	0.0002	7.30 s
4	70%	89.11%	69.76%	78.04%	0.00012	9.87 s

Dari hasil diatas, diperoleh hasil klasifikasi berita online yang cukup baik dengan menggunakan metode *Multivariate Bernoulli*. Secara keseluruhan metode *Multivariate Bernoulli*

dengan maupun tanpa seleksi fitur *Mutual Information* menghasilkan akurasi yang lebih baik dibandingkan metode *Multinomial Naïve Bayes*. Hasil Metode *Multivariate Bernoulli* dengan menggunakan seleksi fitur *Mutual Information* terlihat adanya efisiensi fitur dengan akurasi yang mengalami sedikit penurunan dari sebelum dilakukan seleksi fitur.

10 KESIMPULAN DAN SARAN

Dari hasil penelitian ini dapat disimpulkan bahwa:

1. Algoritma *Multivariate Bernoulli* dapat digunakan dengan cukup baik dalam pengklasifikasian dokumen berita online. Algoritma *Multivariate Bernoulli* memberikan hasil yang lebih baik dibandingkan *Multinomial Naïve Bayes* dengan jumlah *vocabulary* sebanyak 7785 kata. Hasil klasifikasi dokumen diuji menggunakan perhitungan *accuracy, precision, recall* dan *f-measure*. Dari perhitungan tersebut, didapatkan nilai *accuracy* sebesar 80%, *precision* sebesar 94.28%, *recall* sebesar 79.68% dan *f-measure* sebesar 85.08%.
2. Model klasifikasi menggunakan seleksi fitur *Mutual Information* diperoleh hasil maksimal pada algoritma *Multivariate Bernoulli*. Dengan nilai *accuracy* sebesar 10%, *precision* sebesar 33.33%, *recall* sebesar 9.40% dan *f-measure* sebesar 14.35% untuk *Multinomial Naïve Bayes* sedangkan nilai *accuracy* sebesar 70%, *precision* sebesar 89.11%, *recall* sebesar 69.76% dan *f-measure* sebesar 78.04% untuk *Multivariate Bernoulli*. Hasil pengujian mengalami penurunan dikarenakan adanya fitur yang terbuang dari proses seleksi fitur. Meskipun hasil pengujian mengalami penurunan dari hasil pengujian sebelumnya, namun seleksi fitur *Mutual Information* menghasilkan efisiensi jumlah fitur mencapai 52% dari sebelumnya.

Penelitian ini menggunakan data berita online berupa ringkasan dari keseluruhan isi berita. Dengan jumlah *vocabulary* sebanyak 7785 kata dari semua training data yang ada, penelitian selanjutnya dapat dilakukan dengan menggunakan data berita online secara *full-text*, sehingga jumlah *vocabulary* yang didapat lebih optimal.

DAFTAR PUSTAKA

[1] A. Arifin, R. Darwanto, D. A. Navastara and H. T. Ciptaningtyas, "Klasifikasi Online Dokumen Berita dengan Menggunakan Algoritma Suffix Tree Clustering," *In Seminar Sistem Informasi Indonesia (SESINDO2008)*, December 2008.

[2] A. Hamzah, "Klasifikasi Teks dengan Naive Bayes Classifier (NBC) untuk Pengelompokan Teks Berita dan Abstract Akademis," *In Prosiding Seminar Nasional Apikasi Sains & Teknologi (SNAST) Periode III*, pp. p. B269-B277, 2012.

[3] J. Samodra, S. Sumpeno and M. Hariadi, "Klasifikasi Dokumen Teks Berbahasa Indonesia dengan Menggunakan Naive Bayes," *In Seminar Nasional Electrical, Informatics, dan IT's Education*, 2009.

[4] C. Manning, R. P and S. H, Introduction to Information Retrieval, Cambridge University Press, 2009.

[5] A. McCallum and K. Nigam, "A Comparison of Event Models for Naive Bayes Text Classification," *In AAAI-98 workshop on learning for text categorization (Vol. 752, pp. 41-48)*, July 1998.

[6] K.-M. Schneider, "A Comparison of Event Models for Naive Bayes Anti-Spam E-mail Filtering," *In Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, vol. 1, pp. 307-314, April 2003.

[7] J. G. Dimastyo and J. Adisantoso, "Pengukuran Kinerja Spam Filter dengan Feature Selection yang Berbeda Menggunakan Fungsi klasifikasi Multinomial Naïve Bayes," *Makalah Kolokium Ekstensi, 1(1)*, 2014.

[8] R. Imbar, Adelia, M. Ayub and A. Rehatta, "Implementasi Cosine Similarity dan Algoritma Smith-Waterman untuk Mendeteksi Kemiripan Teks," *Jurnal Informatika, 10(1)*, 2014.

[9] A. Sasmoyo, R. Saptono and Wiranto, "Penggunaan Jumlah Frekuensi Kata Terbanyak Sebagai Feature Set Pada Naive Bayes Classifier Untuk Mengklasifikasikan Dokumen Berbahasa Indonesia dan Inggris," *Seminar Nasional Ilmu Komputer*, 2015.

[10] J. Han, M. Kamber and J. Pei, Data Mining: Concepts and Techniques 3rd Edition, Morgan Kaufmann, 2011.

[11] M. Patahuddin, H. Sukoco and A. R. Akbar, "Klasifikasi Web Berdasarkan Domain dan Halaman Utama dengan Algoritme Multinomial Naive Bayes," *In Makalah Seminar Ekstensi (Vol. 1)*, June 2016.

[12] D. M. W. Powers, "Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation," *Journal of Machine Learning Technologies, 37-63*, 2011.