Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 5 Nomor 2, September 2021, pp. 589-600
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

Implementasi *Modified Enhanced Confix Stripping Stemmer* pada Klasifikasi *Fake News* Covid-19

Dyas Rahma Putri[1], Budi Arif Dermawan[2], Intan Purnamasari [3]
[1,2,3] Universitas Singaperbangsa Karawang
Jl. HS. Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang 41361
E-mail : [1]dyas.rahma17087@student.unsika.ac.id, [2]budi.arif@staff.unsika.ac.id,
[3]intan.purnamasari@staff.unsika.ac.id

Abstract

Today's advances in technology and information make communication easier, so that the flow of information can quickly spread. The ease also allows anyone to upload anything on online platforms such as blogs, comments to news articles, social media, etc. that could lead to ambiguity of information or even lead to misleading information. Fake news is information that contains things that are uncertain or not a fact that actually happened. One of the popular news topics nowadays is about the covid-19 virus. This research evaluates the performance of Multinomial Naïve Bayes and Bernoulli Naïve Bayes in conducting fake news classifications related to covid-19. Beside that, we used Modified Enhanced Confix Stripping Stemmer in performing indonesian word standardization that has a variety of shapes and structures. The evaluation showed that Bernoulli Naïve Bayes model had the best performance than Multinomial Naïve Bayes, with the accuracy value of 91%, precision 0.93, recall 0.92, and f-1 score 0.92. In addition, the performance of Modified Enhanced Confix Stripping Stemmer (Modified ECS) algorithm is also perform very well in standardizing words (stemming) Indonesian language.

Keywords*: Classification, Fake news, Modified Enhanced Confix Stripping Stemmer*

Abstrak

Kemajuan teknologi dan informasi saat ini membuat komunikasi semakin mudah, sehingga arus informasi dapat dengan cepat menyebar. Kemudahan tersebut juga membuat siapa saja dapat mengunggah apa pun di platform online seperti blog, komentar ke artikel berita, media sosial, dan lain-lain yang bisa mengakibatkan ambiguitas informasi atau bahkan menimbulkan misleading information. *Fake news* atau berita bohong merupakan informasi yang memuat hal-hal yang tidak pasti atau bukan fakta yang benar-benar terjadi. Salah satu topik berita yang populer saat ini yaitu mengenai virus covid-19. Pada penelitian ini, peneliti menguji performa model *Multinomial Naïve Bayes* dan *Bernoulli Naïve Bayes* dalam melakukan klasifikasi *fake news* terkait covid-19. Penelitian ini menggunakan *Modified Enhanced Confix Stripping Stemmer* dalam melakukan pembakuan kata berbahasa Indonesia yang memiliki beragam bentuk dan struktur imbuhan. Data berita yang digunakan berjumlah 305 data dan diambil dari beberapa situs berita *online*. Hasil evaluasi menunjukkan bahwa performansi model *Bernoulli Naïve Bayes* lebih unggul dibandingkan dari model *Multinomial Naïve Bayes* dengan perolehan nilai akurasi 91%, *precision* 0.93, *recall* 0.92, dan *f-1 score* 0.92. Selain itu, performa Algoritme *Modified Enhanced Confix Stripping Stemmer* (Modified ECS) juga sangat baik dalam melakukan pembakuan kata (*stemming*) berbahasa Indonesia.

Kata kunci*:* Klasifikasi*, Modified Enhanced Confix Stripping Stemmer, Fake News*

1. PENDAHULUAN

Kemajuan teknologi dan informasi saat ini membuat komunikasi semakin mudah, sehingga arus informasi dapat dengan cepat menyebar.

Kemudahan tersebut juga membuat siapa saja dapat mengunggah apa pun di platform *online* seperti *blog*, komentar ke artikel berita, media sosial, dan lain-lain yang bisa mengakibatkan ambiguitas informasi atau bahkan menimbulkan *misleading information*. Tidak dipungkiri, saat ini kebanyakan orang lebih memilih membaca berita melalui media digital ketimbang media cetak [1]. Dengan adanya perubahan pola konsumsi masyarakat dalam memperoleh informasi, sudah sepantasnya media digital dapat menghadirkan berita-berita yang informatif sekaligus meredam kecemasan masyarakat mengenai sebuah isu yang sedang marak dibicarakan ─tidak terkecuali terkait pemberitaan yang populer saat ini─ yaitu mengenai virus covid-19. Pandemi virus covid-19 yang menyerang tidak hanya di Indonesia membuat penyebaran informasi mengenai virus ini tidak dapat terkendali.

Media *online* cenderung lebih cepat dalam memberitakan sebuah kejadian dibandingkan media konvensional, tetapi hal tersebut juga bisa berbahaya apabila kebenarannya belum tervalidasi. *Fake news* atau berita bohong merupakan informasi yang memuat hal-hal yang tidak pasti atau bukan fakta yang benar-benar terjadi [2]. Hal tersebut bisa berbahaya karena dapat menyesatkan persepsi dan anggapan masyarakat terhadap berita yang tidak benar. Maka dari itu, untuk meminimalisir hal tersebut, perlu dikembangkan sebuah metode untuk mengidentifikasi kebenaran suatu berita. Salah satu solusinya yaitu dengan mengklasifikasikan berita tersebut tergolong *fake news* atau bukan.

Pada penelitian sebelumnya, Rahutomo et al. [3] membuat sebuah sistem berbasis PHP-ML untuk mendeteksi berita hoax berbahasa Indonesia dan memanfaatkan Algoritme *Naïve Bayes*. Hasil pengujian secara statis, sistem memperoleh nilai akurasi sebesar 82.6%, sedangkan secara dinamis sistem memperoleh akurasi sebesar 68.3%. Pada penelitian Trisna et al. [4], peneliti melakukan klasifikasi terhadap berita berbahasa Indonesia menggunakan 5 macam *classifier,* diantaranya *Random Forest*, *Multilayer Perceptron, Naïve Bayes, Decision Tree,* dan *Support Vector Machine.* Hasilnya, perpaduan antara TF-IDF dan pembobotan *unigram* & *bigram* menghasilkan kesimpulan bahwa algoritme *Random Forest* mendapatkan nilai optimal diantara algoritme lain dengan nilai akurasi 76.47%, kemudian disusul oleh *Naïve Bayes* dan *Decision Tree* dengan nilai akurasi 74.51%.

Penelitian tentang deteksi *fake news* juga pernah dilakukan oleh Singh et al. [5] dengan melakukan eksperimen terhadap algoritme *Bernoulli Naïve Bayes.* Hasilnya, model *Bernoulli Naïve Bayes* mendapatkan nilai akurasi, *precision, recall* dan *f-1 score* berturut-turut 82.4%, 83.4%, 81,5% dan 82.48%. Selain itu, Yudi & Aditya [6] menerapkan algoritme *Naïve Bayes Classifier* dan *Enhanced Confix Stripping Stemmer* dalam mengklasifikasikan berita menjadi 4 kategori, Olahraga, Teknologi, Ekonomi, dan Lain-lain. Hasil rata-rata akurasinya mencapai 95%.

Terlihat dari beberapa penelitian sebelumnya, *Naïve Bayes Classifer* menjadi salah satu *classifier* yang bekerja cukup baik dalam praktik di banyak domain. Kowsari et al. [7] juga menilai performanya dalam menangani data

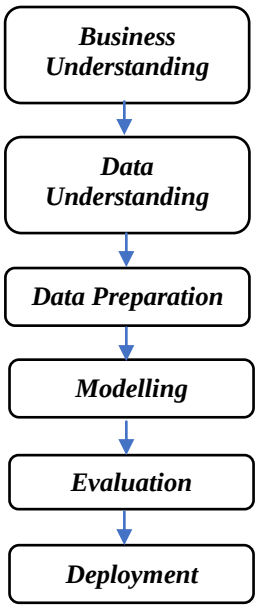
Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 5 Nomor 2, September 2021, pp. 589-600
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

berupa teks cepat dan mudah diimplementasikan. *Naïve Bayes Classifier* memiliki model lain diantaranya, *Multinomial Naïve Bayes, Bernoulli Naïve Bayes* dan *Gaussian Naïve Bayes.* Aggarwal [8] menyatakan bahwa pemrosesan teks berupa dokumen lebih efektif bila menggunakan *Multinomial Naïve Bayes* dan *Bernoulli Naïve Bayes,* karena model tersebut dirancang untuk mengatasi teks yang panjang dan cenderung memiliki kata-kata yang berulang.

Permasalahan yang kerap dialami saat melakukan pemrosesan teks berbahasa Indonesia pada pembelajaran mesin yaitu rumitnya pembakuan kata (*stemming),* karena bahasa Indonesia memiliki beragam bentuk dan struktur imbuhan [9]. Kamus *stemming* bahasa Indonesia telah berkembang seiring berjalannya waktu. Tahitoe & Purwitasari [10] menilai bahwa masih ditemui kesalahan pembakuan kata pada algoritme *Enhanced Confix Stripping Stemmer* (ECS). Oleh karena itu, penelitian ini memanfaatkan algoritme *Modified Enhanced Confix Stripping Stemmer* dalam melakukan pembakuan kata (*stemming)* pada klasifikasi *fake news* covid-19 berbahasa Indonesia dengan membandingkan performa model *Multinomial Naïve Bayes* dan *Bernoulli Naïve Bayes.*

2. METODOLOGI PENELITIAN

Bagian ini menjelaskan secara detail tentang metodologi atau tahapan penelitian yang dilakukan.



Gambar 1. Metdologi Penelitian CRISP-DM

Berdasarkan Gambar 1, penelitian ini mengimplementasikan metodologi CRISP-DM (*Cross-Industry Standard Process for Data Mining*), di mana tahapannya:

Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 5 Nomor 2, September 2021, pp. 589-600
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

2.1. *Business Understanding*

Tahapan ini merupakan proses menentukan inti masalah dan tujuan yang ingin dicapai dalam penelitian. Meliputi studi literatur mengenai teori-teori yang relevan dengan *text mining*, algoritme *Multinomial* & *Bernoulli Naïve Bayes*, juga teknik klasifikasi dokumen berita. Permasalahan yang dianalisis peneliti pada penelitian ini yaitu terkait klasifikasi berita *fake* atau *real* dengan topik covid-19.

2.2. *Data Understanding*

Tahapan selanjutnya adalah memahami dan mengeksplor data dengan terlebih dahulu mengumpulkan data yang dibutuhkan. Penelitian ini memanfaatkan teknik *web scraping*, yaitu teknik pengambilan sejumlah data dengan kata kunci tertentu dari sebuah halaman *website*.

Dataset yang digunakan pada penelitian ini merupakan data berita terkait covid-19 berbahasa Indonesia yang didapatkan dari berbagai macam *website.* Data yang digunakan berjumlah 305 data berita, yang kemudian melewati proses anotasi atau pelabelan data dan digolongkan menjadi kategori *fake* atau *real news.*

2.2.1. *Data preparation*

Tahapan ini merupakan tahap dimana data yang sudah diperoleh dilakukan persiapan dan pembersihan data. Pada penelitian ini atribut yang digunakan hanya isi berita dan label saja, hal ini dimaksudkan untuk mengetahui pola atau karakterikstik berita tiap kategori. Dalam melakukan penelitian, data diolah menggunakan bahasa pemrograman Python. Tahapan *preparation* atau *preprocessing* data diantaranya:

a) *Case folding*

Proses *case folding* merupakan proses penghapusan tanda baca dan mengubah semua kata yang ada dalam dokumen menjadi huruf kecil atau *lower case.* Proses *case folding* juga meliputi pembersihan data karakter yang dinilai tidak berpengaruh, seperti menghapus angka, url, dan spasi berlebih.

Tabel 1. Contoh Penerapan *Case Folding*

Sebelum *case folding*	**Setelah *case folding***
Baru-baru ini obat cina yang bernama Lianhua Qingwen sedang viral di media sosial... (cont)	baru baru ini obat cina yang bernama lianhua qingwen sedang viral di media sosial... (cont)

b) *Tokenization*

Merupakan proses pemisahan atau pemecahan kalimat menjadi kata (token).

Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 5 Nomor 2, September 2021, pp. 589-600
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

Tabel 2. Contoh *Tokenization*

Sebelum *tokenization*	Setelah *tokenization*
baru baru ini obat cina yang bernama lianhua qingwen... (cont)	[‘baru’, ‘baru’, ‘ini’, ‘obat’, ‘cina’, ‘yang’, ‘bernama’, ‘lianhua’, ‘qingwen’, ...(cont)]

c) *Stopword Removal*

Merupakan proses penghapusan kata yang tidak penting dan dibutuhkan dalam sebuah dokumen. Pembuangan kata tersebut bertujuan untuk meminimalisir gangguan saat digunakan sebagai fitur dalam klasifikasi teks. Kata-kata yang termasuk dalam *stopword* yaitu biasanya digunakan untuk menghubungkan kata yang berbeda atau untuk membantu dalam struktur kalimat. Preposisi, konjungsi serta beberapa kata ganti dianggap sebagai *stopword.*

Tabel 3. Contoh Penerapan *Stopword Removal*

Sebelum	Setelah
['setelah', 'kemarin', 'kasus', 'harian', 'covid', 'nyaris', ‘bertambah', ... (cont)]	['kemarin', 'covid', ‘bertambah', ... (cont)]

d) *Stemming*

Yaitu proses pembakuan kata menjadi kata dasar. Tujuan utama *stemming* yaitu untuk mengurangi frekuensi kata turunan. Pada penelitian ini, digunakan algoritme *Modified Enhanced Confix Stripping Stemmer* yang merupakan algoritme perbaikan dari ECS *Stemmer* dan bentuk paling mutakhir dari aturan pemenggalan kata bahasa Indonesia pada *library* Sastrawi.

Tabel 4. Contoh Penerapan *stemming* algoritme *Modified ECS*

Sebelum	Setelah
['pasien', 'dinyatakan', 'sembuh', ... (cont)]	['pasien', 'nyata’, ‘sembuh', ... (cont)]

2.2.2. Ekstraksi Fitur TF-IDF

Salah satu tantangan dalam melakukan pemrosesan teks adalah belajar dari data berdimensi tinggi. Sebab, sebagian besar kata dan frasa dalam sebuah dokumen menyebabkan beban komputasi yang tinggi untuk proses pembelajaran sebuah model. Selain itu, fitur yang tidak relevan dan berlebihan dapat merusak akurasi dan kinerja sebuah model. Jadi, salah satu hal yang dapat dilakukan untuk menanggulangi hal tersebut yaitu menerapkan TF-IDF (*Term Frequency – Inverse Document Frequency*), yang dapat mereduksi fitur untuk mengurangi ukuran fitur teks dan menghindari dimensi ruang fitur yang besar.

Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 5 Nomor 2, September 2021, pp. 589-600
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

Representasi matematis dari bobot istilah dalam dokumen TF-IDF oleh Aggarwal [11] dituliskan pada persamaan (1).

$$tfidf(w) = tf \times \log \frac{N}{df(w)} \quad (1)$$

Di mana tfidf (w) adalah bobot relatif fitur dalam vektor; tf (w) adalah istilah frekuensi (jumlah kemunculan kata dalam dokumen); df (w) adalah frekuensi dokumen (jumlah dokumen yang mengandung kata)dan N adalah banyaknya dokumen dalam korpus.

a) *Modelling*

Tahapan ini merupakan tahap pembuatan model dengan algortime yang ingin digunakan. Penelitian ini menggunakan 2 model klasifikasi yaitu model *Multinomial Naïve Bayes* dan *Bernoulli Naïve Bayes*. Pada tahapan ini, data terlebih dahulu dibagi menjadi *data training* dan *data testing. Data training* merupakan data yang digunakan untuk melatih sebuah model dalam mempelajari pola atau karakteristik sebuah data, sedangkan *data testing* merupakan data yang digunakan untuk menguji model yang sebelumnya telah mengetahui pola dan karakteristik dari sebuah data.

b) *Evaluation*

Merupakan proses pengukuran efektivitas atau performansi model. Penelitian ini menggunakan teknik *confusion matrix* dan *k-fold cross validation* untuk menghitung nilai akurasi, *precision*, *recall,* dan *f1-score.* Konsep perhitungan nilai akuasi, presisi, *recall,* dan *f1-score* merajuk pada Prabhakar et al. [12].

1) *K-fold cross validation*

Teknik *k-fold cross validation* membagi data awal (*data training*) secara acak menjadi k *subset* atau k *folds* yang masing-masing *fold* berjumlah sama. Pelatihan dan pengujian dilakukan sebanyak k kali. Pada iterasi i, *subset* Di dicadangkan sebagai set pengujian, dan *subset* yang tersisa digunakan untuk melatih model. Artinya, pada iterasi pertama, *subset* D2, …, Dk berfungsi sebagai *training set* untuk mendapatkan model pertama, yang diuji pada D1; iterasi kedua dilatih pada *subset* D1, D3, … , Dk dan diuji pada D2; dan seterusnya. Hasil akurasinya merupakan rata-rata dari keseluruhan nilai tiap *folds* [13].

2) *Confusion Matrix*

Pengukuran performa sebuah model dapat dihitung dengan menggunakan *confusion matrix. Confusion Matrix* bekerja dengan menyajikan nilai untuk melihat seberapa baik atau seberapa "akurat" *classifier* dalam memprediksi kelas label [14]. Tabel *confusion matrix* disajikan dalam Tabel 5.

Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 5 Nomor 2, September 2021, pp. 589-600
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

Tabel 5. *Confusion Matrix*

Predicted Class	***True Class***	
	Positive	***Negative***
Positive	TP (*True Positive*)	FP (*False Positive*)
Negative	FN (*False Negative*)	TN (*True Negative*)

3) *Precision*

Precision didefinisikan sebagai perbandingan data yang benar diprediksi positif dengan total kemungkinan positif benar dan salah. Atau dapat juga dikatakan sebagai persentase hasil positif yang relevan. Perhitungan nilai *precision* didapatkan dengan menggunakan rumus pada persamaan (2).

$$Precision = \frac{TP}{(TP+FP)} \quad (2)$$

4) *Recall*

Sedangkan *recall* adalah perbandingan data yang benar diprediksi positif dengan jumlah data yang benar dan salah [15]. Dengan kata lain, *recall* merupakan kasus positif yang diprediksi model dengan benar. Rumus untuk mengetahui nilai *recall* ditunjukkan pada persamaan (3).

$$Recall = \frac{TP}{(TP+FN)} \quad (3)$$

5) *F-1 score*

F-1 score merupakan rata-rata dari *precision* dan *recall* yang tertimbang. Nilai *F-1 score* dapat dihitung berdasarkan persamaan (4).

$$F\text{-}1\ score = \frac{2 \times recall \times precision}{recall+precision} \quad (4)$$

6) *Deployment*

Tahapan ini merupakan tahap pelaporan atau penyampaian hasil berupa representasi dalam bentuk yang mudah dipahami.

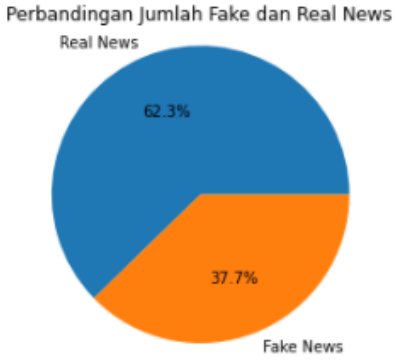
3. HASIL DAN PEMBAHASAN

Pada bagian ini, dipaparkan hasil dari penelitian berdasarkan tahapan yang telah dilakukan. Bagian ini juga akan memperlihatkan visualisasi dari hasil penelitian. Hasil evaluasi merupakan tahap pengukuran performa terhadap model yang telah diterapkan. Nilai yang akan diukur dalam penelitian ini yaitu nilai akurasi, presisi, *recall,* dan *f1-score* dengan menggunakan teknik *confusion matrix* dan *k-fold cross validation.*

Jurnal Sains Komputer & Informatika (J-SAKTI)
Volume 5 Nomor 2, September 2021, pp. 589-600
ISSN: 2548-9771/EISSN: 2549-7200
https://tunasbangsa.ac.id/ejurnal/index.php/jsakti

3.1. Perbandingan proporsi data berita

Gambar 2 menunjukkan visualisasi dari perbandingan jumlah data yang digunakan pada penelitian ini. *Dataset* yang digunakan dibagi menjadi 2 label, yaitu *fake* dan *real news.* Jumlah data *fake news* yaitu sebanyak 115 data, sedangkan data *real news* sebanyak 190 data.



Gambar 2. Perbandingan data *fake* dan *real news*

3.2. Hasil Pembobotan Kata TF-IDF

Tahap ekstraksi fitur merupakan tahap pemberian bobot pada *term* atau kata yang sering muncul dalam dokumen. Tahap ini juga merupakan pengubahan kata yang ada pada dokumen menjadi vektor. Bobot dipengaruhi oleh sering/tidaknya kata tersebut muncul dalam sebuah dokumen.

ahli	air	akal	akibat	...	warga	warkop	warung	waspada
0.000000	0.000000	0.0	0.106144	...	0.000000	0.000000	0.000000	0.000000
0.068555	0.171389	0.0	0.000000	...	0.000000	0.034278	0.034278	0.034278
0.000000	0.000000	0.0	0.087654	...	0.000000	0.000000	0.000000	0.000000
0.000000	0.000000	0.0	0.000000	...	0.118585	0.000000	0.000000	0.000000
0.000000	0.000000	0.0	0.031449	...	0.000000	0.000000	0.000000	0.000000

Gambar 3. Contoh Penerapan TF-IDF

Gambar 3 memperlihatkan hasil vektorisasi yang dilakukan TF-IDF yang berfungsi untuk menghindari dimensi ruang fitur yang besar. Semua *term* di setiap data berita yang terdapat pada korpus, diurutkan berdasarkan abjad dan dihitung kemungkinan munculnya sesuai dengan rumus yang ada. Data yang telah melewati proses pembobotan kata menggunakan TF-IDF, selanjutnya diterapkan model klasifikasi *Multinomial Naïve Bayes* dan *Bernoulli Naïve Bayes.*

3.3. Hasil Evaluasi *10-fold cross validation*

Tabel 6 menunjukkan hasil akurasi performa model *Multinomial Naïve Bayes* dan *Bernoulli Naïve Bayes* dihitung menggunakan teknik *k-fold cross validation* yang jumlah k = 10.

Tabel 6. Hasil Evaluasi *10-fold cross validation*

Fold	*Multinomial NB*	*Bernoulli NB*
1	0.72	1
2	0.63	0.86
3	0.63	0.81
4	0.76	0.85
5	0.85	0.95
6	0.85	0.95
7	0.80	0.90
8	0.71	0.90
9	0.85	1
10	0.85	0.90
Rata-rata	**0.77**	**0.91**

Tabel 6 menunjukkan hasil akurasi kedua model menggunakan teknik *10-fold cross validation.* Rata-rata nilai akurasi dari model *Multinomial Naive Bayes* yaitu 0.77 atau 77%, sedangkan nilai rata-rata akurasi model *Bernoulli Naive Bayes* yaitu 0.91 atau 91%.

3.4. *Confusion Matrix* Model *Multinomial Naïve Bayes*

Tabel 7 menunjukkan hasil *confusion matrix* dari model *Multinomial Naïve Bayes.*

Tabel 7. *Confusion Matrix* Model *Multinomial Naïve Bayes*

Predicted Class	*True Class*	
	Positive	*Negative*
Positive	63	16
Negative	0	13

Hasil *confusion matrix* menunjukkan bahwa nilai *True Positive* (TP) = 63, *True Negatives* (TN) = 13, *False Positives* (FP) = 16 dan *False Negatives* (FN) = 0.

3.5. *Confusion Matrix* Model *Bernoulli Naïve Bayes*

Tabel 8 menunjukkan hasil *confusion matrix* dari model *Bernoulli Naïve Bayes.*

Tabel 8. *Confusion Matrix* Model *Bernoulli Naïve Bayes*

Predicted Class	*True Class*	
	Positive	*Negative*
Positive	58	4
Negative	5	25

Hasil *confusion matrix* menunjukkan bahwa nilai *True Positive* (TP) = 58, *True Negatives* (TN) = 25, *False Positives* (FP) = 4 dan *False Negatives* (FN) = 5.

3.6. Perbandingan Hasil Evaluasi Kedua Model

Tabel 9 menunjukkan perbandingan hasil performansi model *Multinomial Naïve Bayes* dan *Bernoulli Naïve Bayes.*

Tabel 9. Perbandingan Hasil Evaluasi Kedua Model

Performa	*Multinomial NB*	*Bernoulli NB*
Akurasi	0.77	0.91
Precision	0.79	0.93
Recall	1	0.92
F-1 score	0.88	0.92

Pada Tabel 9, terlihat dari hasil pengujian kedua model di tiap skenario pengujian, secara keseluruhan kinerja model *Bernoulli Naïve Bayes* lebih baik dibandingkan dengan *Multinomial Naïve Bayes*. Model *Multinomial Naïve Bayes* memiliki keunggulan pada perolehan nilai *recall* atau *sensitivity,* karena dari ketiga skenario pengujian, semua nilai *recall* nya mendapatkan hasil yang sempurna yaitu 1. Artinya, model dapat memprediksi dengan benar keseluruhan data yang terprediksi positif benar. Nilai akurasi yang didapatkan model *Multinomial Naïve Bayes* yaitu 0.77 atau 77%, *precision* 0.79, *recall* 1, dan *f-1 score* 0.88. Hasil performa terbaik diperoleh oleh model *Bernoulli Naïve Bayes* dengan nilai akurasi 0.91 atau sama dengan 91%, nilai *precision* 0.93, *recall* 0.92, dan *f-1 score* 0.92.

Pada penelitian sebelumnya. Singh et al. [5] melakukan penelitian mengenai deteksi *fake news* menggunakan model *Bernoulli Naïve Bayes* dan menghasilkan performa akurasi 82%. Pada penelitian ini, model *Bernoulli Naïve Bayes* bekerja lebih baik dibandingkan *Multinomial Naïve Bayes* yaitu dengan nilai akurasi terbaik 91%. Beberapa alasan yang dapat mempengaruhinya yaitu model *Bernoulli* dapat dilatih menggunakan lebih sedikit data dan tidak terlalu rentan terhadap *overfitting* sehingga mendapatkan hasil yang lebih optimal. Selain itu, kinerja model *Bernoulli Naïve Bayes* dalam menangani data terbilang sederhana, karena memprediksi kemunculan kata sebagai variabel *boolean* yaitu 0 atau 1.

3.7. Performa Algoritme *Modified Enhanced Confix Stripping Stemmer*

Algoritme *Modified Enhanced Confix Stripping Stemmer* merupakan algoritme perbaikan dari ECS (*Enhaced Confix Stripping Stemmer*) dan bentuk paling mutakhir dari aturan pemenggalan kata bahasa Indonesia pada *library* Sastrawi yang dibuat oleh Tahitoe dan Purwitasari pada tahun 2010. Pada algoritme sebelumnya, yaitu *Enhaced Confix Stripping Stemmer,* masih ditemui kesalahan dalam melakukan pembakuan kata bahasa Indonesia.

Selain itu, *overstemming* dan *understemming* menjadi alasan lain mengapa algoritma ECS *Stemmer* perlu dilakukan perbaikan.

Algoritme *Modified ECS* melakukan beberapa perbaikan pada tabel aturan pemenggalan, menambahkan proses dalam mengurangi sisipan, meningkatkan penyerapan sufiks dalam proses pengurangan sufiks dan menggunakan metode *corpus-based stemming*. Pada penelitian ini, kinerja Algoritme *Modified ECS* menampilkan hasil yang cukup baik dalam melakukan pembakuan kata berbahasa Indonesia. Karena berhasil melakukan pembakuan kata, seperti kata 'mendonorkan' menjadi 'donor', 'terkonfirmasi' menjadi 'konfirmasi', 'perekonomian' menjadi 'ekonomi', dan lain-lain. Namun ada beberapa kata yang tidak seharusnya melewati proses pembakuan kata seperti kata 'kegiatan' berubah menjadi 'giat', kata 'gerakan' berubah menjadi 'gera' dan lainnya. Namun, hal tersebut tidak mengurangi efektifitas performa algoritme *Modifies ECS.* Terbukti, kinerja model yang diterapkan pada *dataset* memberikan hasil yang optimal, tentunya dipengaruhi pula oleh algoritme *stemming Modified ECS.*

4. SIMPULAN

Penerapan model *Bernoulli Naïve Bayes* pada klasifikasi *fake news* covid-19 berbahasa Indonesia dengan TF-IDF dan algoritme *Modified Enhanced Confix Stripping Stemmer* menghasilkan performa yang optimal jika dibandingkan dengan model klasifikasi *Multinomial Naïve Bayes.* Hasil performa terbaik dari model *Bernoulli Naïve Bayes* memperoleh nilai akurasi sebesar 91%, *precision* 0.93, *recall* 0.92, *f-1 score* 0.92. Sedangkan performa *Multinomial Naïve Bayes* memperoleh hasil akurasi 77%, *precision* 0.79, *recall* 1, dan *f-1 score* 0.88. Selain itu, Algoritme *Modified Enhanced Confix Stripping Stemmer* (*Modified* ECS) bekerja sangat baik dalam melakukan pembakuan kata berbahasa Indonesia.

DAFTAR PUSTAKA

[1] B. D. Wicaksono, "IMR 2019: 5 Fakta Perubahan Pola Konsumsi Media Millennial," 2019. https://www.idntimes.com/tech/trend/bayu/survei-ims-2019-5-fakta-perubahan-pola-konsumsi-media-millennial/5.

[2] A. Prasetyo, B. D. Septianto, G. F. Shidik, and A. Z. Fanani, "Evaluation of feature extraction TF-IDF in Indonesian hoax news classification," in *Proceedings - 2019 International Seminar on Application for Technology of Information and Communication: Industry 4.0: Retrospect, Prospect, and Challenges, iSemantic 2019*, 2019, pp. 1–6, doi: 10.1109/ISEMANTIC.2019.8884291.

[3] F. Rahutomo, I. Yanuar, R. Pratiwi, and D. M. Ramadhani, "Eksperimen Naive Bayes pada Deteksi Berita Hoax Berbahasa Indonesia," *J. Penelit. Komun. dan Opini Publik V*, vol. 23, no. 1, pp. 1–15, 2019.

[4] T. Trisna *et al.*, "Analysis and Detection of Hoax Contents in Indonesian News Based on Machine Learning," *JIPN (Journal Informatics Pelita*

Nusantara), vol. 4, no. 1, 2019.
[5] M. Singh, M. Wasim Bhatt, H. S. Bedi, and U. Mishra, "Performance of bernoulli's naive bayes classifier in the detection of fake news," *Mater. Today Proc.*, no. xxxx, 2020, doi: 10.1016/j.matpr.2020.10.896.
[6] E. Yudi and M. Aditya, "Klasifikasi Dokumen Berita Menggunakan Algoritma Enhanced Confix Stripping Stemmer dan Naïve Bayes Classifier," *J. Nas. Teknol. dan Sist. Inf.*, vol. 02, pp. 90–99, 2020.
[7] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey," *MDPI*, vol. 10, no. 4, pp. 1–68, 2019, doi: 10.3390/info10040150.
[8] C. C. Aggarwal, *Data Mining text book*, vol. 53, no. 9. 2015.
[9] Y. N. Fadziah and E. F. R, "Penerapan Algoritma Enchanced Confix Stripping dalam Pengukuran Keterbacaan Teks Menggunakan Gunning Fog Index," *JATIKOM J. Teor. dan Apl. Ilmu Komput.*, vol. 1, no. 1, pp. 15–24, 2018.
[10] A. D. Tahitoe and D. Purwitasari, "Implementasi Modifikasi Enhanced Confix Stripping Stemmer Untuk Bahasa Indonesia dengan Metode Corpus Based Stemming," pp. 1–15, 2010.
[11] C. C. Aggarwal, *Mining Text Data*. Springer, 2012.
[12] R. Prabhakar, D. C. Manning, and H. Schütze, *Introduction to Information Retrieval*. 2008.
[13] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, vol. 54, no. Second Edition. 2006.
[14] J. Han, M. Kamber, and J. Pei, *Data mining: Data mining concepts and techniques*. 2014.
[15] G. Miner, J. Elder, R. A. Nisbet, J. Thompson, and R. Foley, *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*, 1st ed. Elsevier Ltd, 2012.