

DETEKSI HOAX PADA BERITA ONLINE BAHASA INGGRIS MENGGUNAKAN *BERNOULLI NAÏVE BAYES* DENGAN EKSTRAKSI FITUR TF-IDF

Agri Yodi Prayoga, Asep Id Hadiana, Fajri Rakhmat Umbara

Universitas Jenderal Achmad Yani (UNJANI) Jawa Barat, Indonesia

Email: agriyodip17@if.unjani.ac.id, asephadiana@lecturer.unjani.ac.id,

fajri.umbara@lecturer.unjani.ac.id

INFO ARTIKEL

Diterima
25 September 2021
Direvisi
05 Oktober 2021
Disetujui
15 Oktober 2021

Kata Kunci: *tf-idf*;
bernoulli's naïve bayes;
text preprocessing;
confusion matrix

ABSTRAK

Fenomena yang disebut sebagai "berita palsu" saat ini mengacu pada publikasi online dari pernyataan fakta palsu yang disengaja. Tujuan pembuatan berita hoax adalah untuk mempengaruhi pembaca berita untuk mencegah tindakan yang benar. Deteksi berita hoax ini berperan penting bagi pemerintah dan masyarakat, sebab itu berita hoax harus segera dideteksi untuk menghindari efek yang dapat ditimbulkannya. Penelitian ini bertujuan untuk mengetahui performa dari penggunaan algoritma *Bernoulli Naïve Bayes* dengan ekstraksi fitur TF-IDF dalam mendeteksi berita hoax. Tahapan penelitian ini disusun sebagai berikut, yaitu pengumpulan data dan *labeling*, *text preprocessing*, ekstraksi fitur dengan TF-IDF, pembagian *dataset*, *classification*, *Output* (Klasifikasi *Fake/True*), evaluasi, dan kesimpulan. Hasil implementasi menunjukkan model prediksi yang dibangun dengan 8800 data berita, mampu menghasilkan nilai akurasi sebesar 98,5% dari jumlah data uji sebanyak 2.200 data berita, dimana akurasi dari prediksi model untuk label 1 (*Fake*) sebesar 97,8%, dan akurasi untuk label 0 (*True*) sebesar 99,1%, diikuti dengan nilai *precision* 99,1%, *recall* 97,8%, dan *f1-score* 98,4%. Dari implementasi tersebut penggunaan ekstraksi fitur TF-IDF pada algoritma *Bernoulli Naïve Bayes* mampu meningkatkan akurasi sebesar 16,08%, *precision* 15,7%, *recall* 16,22%, dan *f1-score* 15,92% bila dibandingkan dengan hasil penelitian sebelumnya.

ABSTRACT

The phenomenon referred to as "fake news" today refers to the online publication of deliberately false statements of fact. The purpose of making hoax news is to influence news readers to prevent the right action. Detection of hoax news plays an important role for the government and society, therefore hoax news must be detected immediately to avoid the effects it can cause. This study aims to determine the performance of using the

How to cite:

Prayoga, A. Y., Hadiana, A. I., Umbara, F. R., (2021) Deteksi Hoax pada Berita Online Bahasa Inggris Menggunakan Bernoulli Naïve Bayes dengan Ekstraksi Fitur TF-IDF. *Jurnal Syntax Admiration* 2(10). <https://doi.org/10.46799/jsa.v2i10.327>
2722-5356

E-ISSN:

Published by:

Ridwan Institute

Bernoulli Naïve Bayes algorithm with TF-IDF feature extraction in detecting hoax news. The stages of this research are structured as follows, namely data collection and labeling, text preprocessing, feature extraction with TF-IDF, dataset distribution, classification, Output (Fake/True Classification), evaluation, and conclusions. The implementation results show that the prediction model built with 8800 news data, is able to produce an accuracy value of 98.5% of the total test data of 2,200 news data, where the accuracy of the prediction model for label 1 (Fake) is 97.8%, and the accuracy for label 1 (Fake) is 97.8%. label 0 (True) is 99.1%, followed by precision value 99.1%, recall 97.8%, and f1-score 98.4%. From this implementation, the use of TF-IDF feature extraction on the Bernoulli Naïve Bayes algorithm is able to increase accuracy by 16.08%, precision 15.7%, recall 16.22%, and f1-score 15.92% when compared to the results of previous studies.

Keywords: *tf-idf; bernoulli's bayes; preprocessing; confusion matrix*

Pendahuluan

Di seluruh dunia, pengaruh berita palsu yang semakin meningkat dirasakan setiap hari mulai dari politik, pendidikan hingga pasar keuangan (Ajao et al., 2018). Secara historis, akarnya kembali ke abad ke-17 yang berawal dalam bentuk “Propaganda” yang diubah menjadi “Misinformasi” di era Perang Dingin (Gravanis et al., 2019). Fenomena yang disebut sebagai “berita palsu” saat ini mengacu pada publikasi *online* dari pernyataan fakta palsu yang disengaja (Gravanis et al., 2019). Pembuatan berita *hoax* bertujuan untuk mempengaruhi pembaca berita untuk melakukan hal-hal yang bertentangan atau mencegah tindakan yang benar (Rahutomo et al., 2019). Salah satu contoh paling mencolok tentang bagaimana berita palsu dapat memengaruhi opini publik adalah penyebaran berita palsu melalui *platform* media sosial pada saat kampanye presiden AS pada tahun 2016 (Bondielli & Marcelloni, 2019). Penyebarannya begitu cepat karena berita palsu yang tersebar luas sangat sulit untuk ditangani di dunia digital saat ini, dimana ada ribuan *platform* berbagi informasi yang dapat digunakan untuk menyebarkan berita palsu atau misinformasi (Agarwal et al., 2019), sedangkan sebelum munculnya internet, jurnalis ditugaskan untuk melakukan verifikasi dan pengecekan fakta dari berita dan sumbernya, sehingga membuat opini publik terhadap berita palsu semakin terbatas (Bondielli & Marcelloni, 2019). Untuk itu berita *hoax* ini harus segera dideteksi untuk menghindari efek yang dapat ditimbulkannya. Deteksi ini penting tidak hanya untuk perusahaan dan media saja namun bagi pemerintah dan masyarakat juga, dimana memungkinkan pemerintah mengurangi bahkan menghilangkan penyebaran berita *hoax* tersebut. Dalam penelitian sebelumnya yang telah mengidentifikasi metode terkini seputar isu-isu kritis dan tren masa depan dalam mendeteksi berita palsu dan rumor (Bondielli & Marcelloni, 2019), dijelaskan bahwa penelitian perihal pendeteksian berita palsu terus meningkat dari tahun ke tahun yang dapat diartikan upaya untuk menghilangkan penyebaran berita palsu ini tidak pernah berhenti dan terus dilakukan. Dalam penelitian tersebut (Bondielli &

Marcelloni, 2019), disebutkan secara khusus sebagian besar metode yang digunakan untuk melakukan deteksi berita palsu ialah menggunakan pendekatan *machine learning* dengan strategi pembelajaran yang diawasi (*supervised learning strategy*) (Bondielli & Marcelloni, 2019). Algoritma *machine learning* yang dimaksud seperti *Support Vector Machine* (Gravanis et al., 2019), *Stochastic Gradient Descent* (Prasetijo et al., 2017), *Naïve Bayes* (Granik & Mesyura, 2017), C45 (Prasetijo et al., 2017) dan K-NN (Ahmed et al., 2018).

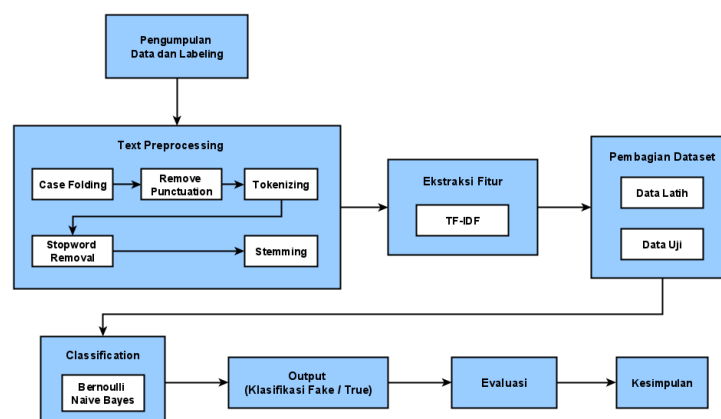
Beberapa penelitian yang mendekati penelitian ini diantaranya penelitian Singh dkk. (G. Singh et al., 2019) yang melakukan klasifikasi artikel berita tekstual pada banyak peristiwa penting yang terjadi di India pada tahun 2018 untuk memprediksi apakah sentimen artikel berita yang digunakan positif atau negatif menggunakan dua pendekatan yaitu *Multivariate Bernoulli Naïve Bayes* dan *Multinomial Naïve Bayes*, hasil yang didapatkan *Multivariate Bernoulli Naïve* adalah 69,15%. Kemudian pada penelitian Prasetyo dkk. (Prasetyo et al., 2019) melakukan evaluasi kinerja ekstraksi fitur TF-IDF yang dikombinasikan dengan metode klasifikasi seperti LSVM, MNB, k-NN, dan Klasifikasi logistik untuk identifikasi berita hoax Indonesia. Dataset berita hoax dikumpulkan dari Dinas Komunikasi dan Informatika Provinsi Jawa Tengah dan www.turnbackhoax.id, akurasi yang didapatkan *Multinomial Naïve Bayes (MNB)* adalah 77, 33%. Pada penelitian (Rahutomo et al., 2019) Rahutomo dkk. melakukan eksperimen klasifikasi *Naïve Bayes* pada deteksi berita *hoax* berbahasa Indonesia, dengan membangun sistem untuk mengklasifikasikan berita daring berbahasa Indonesia dengan ekstraksi fitur *term frequency (TF)* dan algoritma klasifikasi *Naïve Bayes* dengan menggunakan komponen *library PHP-Machine Learning*. Menghasilkan rata-rata akurasi tertinggi sekitar 83%. Kemudian Poddar dkk. (Poddar et al., 2019), melakukan deteksi berita palsu dengan menggunakan model komputasi pembelajaran mesin probabilistik dan geometris, dimana pada penelitian tersebut membandingkan dua *vectorizer teks* yaitu *Count Vectorizer* dan TF-IDF untuk menemukan *vectorizer* yang sesuai untuk masing-masing algoritma yang dipakai, algoritma tersebut yaitu *Naive Bayes*, *Support Vector Machine (SVM)*, regresi logistik, dan *Decision Tree*. Hasil yang didapatkan algoritma *Naïve Bayes* yaitu dengan menggunakan *Count Vectorizer* sekitar 86,3% dan menggunakan TF-IDF sekitar 85,4%. Lalu pada penelitian (M. Singh et al., 2020) membahas algoritma pembelajaran mesin bernama *Bernoulli's Naive Bayes Classifier* untuk mendeteksi berita palsu. Akurasi yang didapatkan algoritma ini sekitar 83% yang dibandingkan dengan algoritma *Gaussian Naive Bayes* yang memperoleh tingkat akurasi sekitar 72%, dimana kedua algoritma tersebut diterapkan pada *dataset* yang sama (M. Singh et al., 2020). dan pada penelitian (Wongso et al., 2017) yang bertujuan untuk menemukan algoritma yang tepat dalam mengklasifikasikan artikel berita dalam Bahasa Indonesia. Akurasi yang didapatkan kombinasi TFIDF dan MNB sebesar 98,4% mengungguli kombinasi TFIDF dan BNB yaitu 98,2% (Mulyani et al., 2021).

Dari penelitian sebelumnya ini, Ekstraksi fitur TF-IDF berfungsi dengan baik untuk meningkatkan *recall* dan presisi dalam melakukan representasi vektor teks,

dimana TF-IDF juga mengungguli ekstraksi fitur *Term Frequency (TF)*, dalam mengurangi ukuran fitur teks dalam menghindari dimensi ruang fitur yang besar, karena fitur yang tidak relevan atau berlebihan dapat merusak keakuratan dari kinerja *classifier*. Kemudian *Bernoulli Naïve Bayes* juga berfungsi dengan baik dalam mengklasifikasikan suatu arikel berita yang dibuktikan pada penelitian terdahulu, dimana algoritma ini merupakan versi perluasan dari *Multinomial Naive Bayes* dengan prediktor sebagai variabel Boolean yaitu 0 dan 1 dalam mendeteksi berita palsu. Maka dari itu penelitian ini mengusulkan deteksi berita hoax menggunakan algoritma *Bernoulli Naïve Bayes* dengan ekstraksi fitur TF-IDF untuk mengetahui performa dari kombinasi tersebut dalam mendeteksi berita hoax (Budiman et al., 2021). Penerapan akan dilakukan pada *dataset fake news* dan *real news* yang sudah dikumpulkan. Kemudian kombinasi algoritma ini akan dievaluasi performanya dengan cara menganalisa nilai *confusion matrix* yang didapatkan, analisa yang dimaksud yaitu menghitung nilai akurasi, *recall*, presisi, dan *f1-score*. Hal tersebut dilakukan demi mencapai tujuan penelitian yang telah dibuat yaitu mengetahui performa dari algoritma *Bernoulli Naïve Bayes* dengan ekstraksi fitur TF-IDF dalam mendeteksi berita *hoax*. Adapula manfaat dari penelitian ini yaitu kombinasi dari algoritma *machine learning* dengan ekstraksi fitur yang digunakan, dapat dijadikan sebagai referensi bagi penelitian selanjutnya dalam mendeteksi berita *hoax*.

Metode Penelitian

Dalam melakukan tahapan yang sudah disiapkan, implementasi akan dibantu dengan penggunaan *library machine learning* bernama *scikit-learn* pada pemrograman *python*. Tahapan dalam penelitian ini disusun sebagai berikut, yaitu : pengumpulan data dan *labeling*, *text preprocessing*, ekstraksi fitur, pembagian *dataset*, *classification*, *Output* (Klasifikasi *Fake/True*), evaluasi, dan kesimpulan. Tahapan tersebut ditunjukkan pada Gambar 1.



Gambar 1
Tahapan Penelitian

Tahap *pertama* yaitu Pengumpulan data dan *labeling*, data yang akan digunakan berasal dari website Kaggle dimana data tersebut dikumpulkan menggunakan teknik *crawling* pada portal berita online berbahasa Inggris. Terdiri dari dataset *fake news* dan *dataset real news*. Masing-masing dataset ini memiliki 20,000 lebih data berita, tepatnya dataset *fake news* sebanyak 23,489 ribu data, dan dataset *real news* sebanyak 21,418 ribu data. Masing-masing dataset terdiri dari 4 atribut yaitu *title*, *text*, *subject*, dan *date*. Kemudian untuk pemberian label akan menggunakan nilai *biner* yaitu 1 untuk *fake news* dan 0 untuk *real news*.

Tahap *Kedua* yaitu melakukan *Text preprocessing*, pada tahap ini akan dilakukan proses *case folding*, *tokenizing*, *stopword removal*, dan *stemming* pada *dataset*. Hal tersebut untuk membantu mengurangi ukuran data aktual dengan menghapus informasi tidak relevan yang ada dalam data.

Tahap *Ketiga* yaitu melakukan ekstraksi fitur dengan TF-IDF, Tahap ini akan melakukan vektorisasi teks pada tiap dokumen dengan TF-IDF.

Tahap *Keempat* yaitu pembagian dataset, Pada tahap ini dataset yang digunakan akan dibagi menjadi dua bagian yaitu data latih dan data uji. Data latih akan digunakan sebagai data untuk membentuk/melatih model *Bernoulli Naïve Bayes*, sedangkan data uji digunakan untuk menguji model yang telah dibentuk. Pembagian dataset tersebut akan menggunakan perbandingan 80:20, yang artinya 80% data pada dataset akan dijadikan data latih (*train data*) kemudian sisa datanya yaitu 20% akan dijadikan sebagai data uji (*test data*).

Tahap *Kelima* yaitu melakukan *Classification*, Pada tahap ini akan dilakukan proses klasifikasi data dengan algoritma *Bernoulli Naïve Bayes*, dimana data latih yang sudah dibagi sebelumnya akan digunakan untuk membentuk model, kemudian setelah model terbentuk barulah data uji akan digunakan untuk menguji model tersebut. Dalam tahap pembentukan model *Bernoulli Naïve Bayes* akan dilakukan pengevaluasian kinerja model dengan metode *K-Fold Cross Validation*.

Tahap *Keenam* yaitu *Output* (Klasifikasi *Fake/True*), Pada tahap ini model akan mengeluarkan hasil klasifikasi dari data uji, hasil klasifikasi tersebut berupa mana dokumen yang terklasifikasi 1 (*Fake*) dan mana dokumen yang terklasifikasi 0 (*True*).

Tahap *Ketujuh* yaitu Evaluasi, tahap ini akan melakukan evaluasi dengan cara menganalisa nilai confusion matrix untuk mendapatkan nilai akurasi, *recall*, presisi, dan *f1-score* dari hasil pengujian model tersebut.

Tahap *Kedelapan* yaitu Kesimpulan, tahap ini akan menyimpulkan bagaimana performa dari algoritma *Bernoulli Naïve Bayes* dan TF-IDF dalam mendeteksi berita hoax.

Hasil dan Pembahasan

1. Data Preparation

Pada penelitian ini, tahapan dari *Data Preparation* akan dibantu dengan penggunaan *library python* bernama *pandas* yang dioperasikan pada *tool* bernama *jupyter notebook*. Dataset yang akan digunakan pada penelitian ini berasal dari

website Kaggle dimana data tersebut dikumpulkan menggunakan teknik *crawling* pada portal berita online berbahasa Inggris. Terdiri dari dataset *fake news* dan dataset *real news*. Masing-masing dataset ini memiliki jumlah data yaitu dataset *fake news* sebanyak 23,489 ribu data, dan dataset *real news* sebanyak 21,418 ribu data, berisi 4 kolom yaitu *title*, *text*, *subject*, dan *date*. Penampakan dari dataset tersebut dapat dilihat pada Tabel 1.

Tabel 1
Dataset Fake News

<i>Title</i>	<i>Text</i>	<i>Subject</i>	<i>Date</i>
Donald Trump Sends Out Embarrassing New Year's Eve Message; This is Disturbing	Donald Trump just couldn't wish all Americans a Happy New Year and leave it at that. Instead, he had to give a shout out to his enemies, haters and the very dishonest fake news media. The former reality show star had just one job to do and he couldn't do it. As our Country rapidly grows stronger and smarter, I want to wish all of my friends, supporters, enemies, haters, and even the very dishonest Fake News Media, a Happy and Healthy New Year, President Angry Pants tweeted. 2018 will be a great year for America! As our Country rapidly grows stronger and smarter, I want to wish all of my friends, supporters, enemies, haters, and even the very dishonest Fake News Media, a Happy and Healthy New Year. 2018 will be a great year for America!	News	December 31, 2017

2. Data Integration

Pada tahap ini langkah yang akan dilakukan yaitu mengintegrasikan beberapa file, karena dalam penelitian ini menggunakan 2 dataset yang sumbernya berbeda yaitu *fake news* dan *real news* maka kedua dataset tersebut perlu diintegrasikan satu sama lain. Terutama menambahkan kolom baru bernama "*article*" yang merupakan gabungan dari kolom "*title*" yang merupakan judul berita dan "*text*" yang merupakan isi berita, serta menambahkan juga kolom bernama "*fake*" sebagai label *supervise* untuk setiap dokumen berita yang diisi dengan nilai *binary* (1 dan 0), nilai 1 menandakan bahwa berita tersebut *fake* dan nilai 0 menandakan dok. berita tersebut *real*. Sebagai contoh akan diambil dua data berita dari dataset tersebut, lalu untuk hasil proses *Data Integration* dapat dilihat pada Tabel 2.

Tabel 2
Dataset setelah proses Data Integration

<i>Title</i>	<i>Text</i>	<i>Subject</i>	<i>Date</i>	<i>Fake</i>	<i>Article</i>
<i>Donald Trump Sends Out Embarrassing New Year's Eve Message; This is Disturbing</i>	<i>Donald Trump just couldn t wish all Americans a Happy New Year and leave it at that. Instead, he had to give a shout out to his enemies, haters and the very dishonest fake news media. The former reality show star had just one job to do and he couldn t do it. As our Country rapidly grows stronger and smarter, I want to wish all of my friends, supporters, enemies, haters, and even the very dishonest Fake News Media, a Happy and Healthy New Year, President Angry Pants tweeted.</i>	<i>News</i>	<i>December 31, 2017</i>	<i>1</i>	<i>Donald Trump Sends Out Embarrassing New Year's Eve Message; This is Disturbing Donald Trump just couldn t wish all Americans a Happy New Year and leave it at that. Instead, he had to give a shout out to his enemies, haters and the very dishonest fake news media. The former reality show star had just one job to do and he couldn t do it. As our Country rapidly grows stronger and smarter, I want to wish all of my friends, supporters, enemies, haters, and even the very dishonest Fake News Media, a Happy and Healthy New Year, President Angry Pants tweeted.</i>

3. Data Reduction

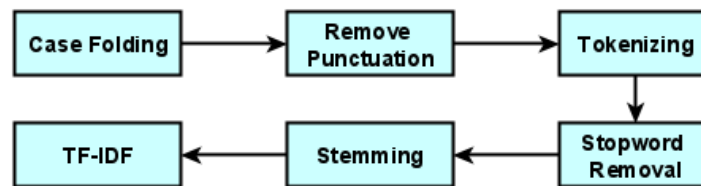
Setelah dilakukan proses *Data Integration*, total data berita pada dataset menjadi 44,907 data berita. Untuk itu pada tahapan ini akan dilakukan proses *Data Reduction* atau mereduksi jumlah data pada dataset, yang bertujuan untuk mengurangi beban komputasi dari proses data mining pada perangkat keras yang digunakan. Dataset tersebut akan direduksi jumlah datanya sekitar 75% sehingga data tersisa 25% atau sekitar 11.000 data berita, yang terdiri dari 5500 berita *fake* dan 5500 berita *real*.

4. Data Cleaning

Pada tahap ini akan dilakukan penghapusan redudansi data berita (data duplikat).

5. Data Transformation

Pada tahap ini dataset yang telah diperoleh setelah proses *Data Cleaning* akan diubah supaya dataset tersebut dapat diproses oleh algoritma data mining yang digunakan. Data yang akan ditransformasikan ialah data pada kolom "*article*", kolom "*article*" dipilih karena kolom tersebut memiliki bagian dari suatu artikel berita yaitu judul (*title*) dan isi berita (*text*). Tahapan *Data Transformation* pada penelitian ini ditunjukkan pada Gambar 2.



Gambar 2
Alur Data Transformation

6. Case Folding

Pada tahap ini semua huruf yang ada pada data artikel akan dirubah menjadi huruf kecil (*lowercase*). Sebagai contoh, diambil satu data berita dari dataset tersebut. Hasil dari proses *Case Folding* pada data artikel dapat dilihat pada Tabel 3.

Tabel 3
Hasil Proses *Case Folding* Pada Kolom *Article*

<i>Article</i>	Hasil <i>Case Folding</i>
<i>Donald Trump Sends Out Embarrassing New Year's Eve Message; This is Disturbing Donald Trump just couldn t wish all Americans a Happy New Year and leave it at that. Instead, he had to give a shout out to his enemies, haters and the very dishonest fake news media. The former reality show star had just one job to do and he couldn t do it.</i>	<i>donald trump sends out embarrassing new year's eve message; this is disturbing donald trump just couldn t wish all americans a happy new year and leave it at that. instead, he had to give a shout out to his enemies, haters and the very dishonest fake news media. the former reality show star had just one job to do and he couldn t do it.</i>

7. Remove Punctuation

Setelah proses *Case Folding* dilakukan, selanjutnya pada tahap *Remove Punctuation* semua tanda baca dan angka akan dihapus dari artikel berita. List dari tanda baca yang akan dihapus diantaranya seperti berikut `["#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~]`. Pada tahap ini juga semua kata yang mengandung unsur url seperti `http`, `https` dan kata spesial yang peneliti masukan yaitu `bit.ly`, dan `twitter.com` akan dihapus dari data berita. Hasil dari tahapan tersebut dapat dilihat pada Tabel 4.

Tabel 4
Hasil Proses *Remove Punctuation*

Hasil <i>Case Folding</i>	Hasil <i>Remove Punctuation</i>
<i>1815onald trump sends out embarrassing new year's eve message; this is disturbing 1815onald trump just couldn t wish all americans a happy new year and leave it at that. Instead, he had to give a shout out to his enemies, haters and the very dishonest fake news media. The former reality show star had just one job to do and he couldn t do it.</i>	<i>1815onald trump sends out embarrassing new years eve message this is disturbing 1815onald trump just couldn t wish all americans a happy new year and leave it at that instead he had to give a shout out to his enemies haters and the very dishonest fake news media the former reality show star had just one job to do and he couldn t do it.</i>

8. Tokenizing

Pada tahap ini semua kalimat pada setiap artikel berita akan dipotong menjadi beberapa kata. Sebagai contoh untuk melakukan tahapan *tokenizing* akan digunakan sebagian data dari hasil *Remove Punctuation* pada tahap sebelumnya. Lalu untuk hasil dari proses *Tokenizing* dapat dilihat pada Tabel 5.

Tabel 4
Hasil Tokenizing

Hasil Remove Punctuation	Hasil Tokenizing
<i>donald trump sends out embarrassing new years eve message this is disturbing donald trump just couldn t wish all americans a happy new year and leave it at that instead he had to give a shout out to his enemies haters and the very dishonest fake news media the former reality show star had just one job to do and he couldn t do it.</i>	<i>1816onald, trump, sends, out, embarrassing, new, years, eve, message, this, is, disturbing, 1816onald, trump, just, couldn, t, wish, all, americans, a, happy, new, year, and, leave, it, at, that, instead, he, had, to, give, a, shout, out, to, his, enemies, haters, and, the, very, dishonest, fake, news, media, the, former, reality, show, star, had, just, one, job, to, do, and, he, couldn, t, do, it.</i>

9. Stopword Removal

Setelah proses *tokenizing* setiap kata akan dicek satu per satu apakah termasuk ke dalam list *stopword* dalam bahasa inggris atau tidak, bila termasuk pada list *stopword* maka kata tersebut akan dihapus atau dihilangkan. Hasil dari proses *Stopword Removal* dapat dilihat pada Tabel 6.

Tabel 5
Hasil Stopword Removal

Hasil Tokenizing	Hasil Stopword Removal
<i>donald, trump, sends, out, embarrassing, new, years, eve, message, this, is, disturbing, donald, trump, just, couldn, t, wish, all, americans, a, happy, new, year, and, leave, it, at, that, instead, he, had, to, give, a, shout, out, to, his, enemies, haters, and, the, very, dishonest, fake, news, media, the, former, reality, show, star, had, just, one, job, to, do, and, he, couldn, t, do, it.</i>	<i>Donald, trump, sends, embarrassing, new, years, eve, message, disturbing, donald, trump, wish, americans, happy, new, year, leave, instead, give, shout, enemies, haters, dishonest, fake, news, media, former, reality, show, star, one, job, country, rapidly, grows, stronger, smarter, want, wish, friends, supporters, enemies, haters, even, dishonest, fake, news, media.</i>

10. Stemming

Pada tahap ini imbuhan akhir pada setiap kata akan dipotong, tujuannya agar didapatkan bentuk dasar dari kata tersebut. Algoritma *Stemming* yang akan digunakan pada penelitian ini ialah algoritma *PorterStemmer* untuk bahasa inggris, dalam algoritma tersebut imbuhan akhir yang dipotong diantaranya *ed*, *es*, *s*, dan *ing*. Hasil dari proses *Stemming* dapat dilihat pada Tabel 7.

Tabel 6
Hasil Stemming

Hasil Stopword Removal	Hasil Stemming
<i>donald, trump, sends, embarrassing, new, years, eve, message, disturbing, donald, trump, wish, americans, happy, new, year, leave, instead, give, shout, enemies, haters, dishonest, fake, news, media, former, reality, show, star, one, job, country, rapidly, grows, stronger, smarter, want, wish, friends, supporters, enemies, haters, even, dishonest, fake, news, media.</i>	<i>donald, trump, send, embarrass, new, year, eve, messag, disturb, donald, trump, wish, american, happi, new, year, leav, instead, give, shout, enem, hater, dishonest, fake, news, media, former, realiti, show, star, one, job, countri, rapidli, grow, stronger, smarter, want, wish, friend, support, enemi, hater, even, dishonest, fake, news, media.</i>

11. Ekstraksi Fitur dengan TF-IDF

Tahap selanjutnya yaitu melakukan vektorisasi teks pada setiap dokumen berita. Terdapat beberapa tahapan yang harus dilakukan dalam proses perhitungan TF-IDF, yaitu melakukan preprocessing text, menghitung jumlah keseluruhan dokumen, menghitung jumlah term (kata) pada semua dokumen, dan menghitung bobot term pada masing-masing dokumen. Berikut ini contoh perhitungan untuk melakukan vektorisasi teks menggunakan metode TF-IDF.

Sebagai contoh dokumen berita yang akan digunakan, yaitu:

Dokumen 1 : Racist Chicago Cop Who Beat A Handcuffed Black Cop

Dokumen 2 : People Catch Thieves in Alabama Area

Dokumen 3 : local police catch the thief

12. Melakukan Preprocessing Text

Sebelum melakukan vektorisasi teks, setiap dokumen akan dilakukan *preprocessing text* seperti pada tahapan sebelumnya yaitu *case folding* (merubah semua huruf menjadi *lowercase*), *remove punctuation* (menghapus tanda baca), *tokenizing* (memotong kalimat menjadi beberapa token kata), menghapus *stopword* dalam bahasa inggris, dan *stemming* (merubah kata berimbuhan menjadi kata dasarnya). Hasil *preprocessing text* pada ketiga dokumen tersebut seperti berikut:

Dokumen 1 = *racist chicago cop beat black cop*

Dokumen 2 = *people catch thief alabama area*

Dokumen 3 = *local police catch thief*

13. Menghitung Jumlah Keseluruhan Dokumen

Kemudian jumlah keseluruhan dokumen akan dihitung, pada contoh diatas total dokumen adalah tiga maka nilai (D) = 3.

14. Menghitung Jumlah Term pada Semua Dokumen

Selanjutnya menghitung jumlah *term* (kata) pada semua dokumen. Jumlah term yang didapatkan ditunjukkan pada Tabel 8.

Tabel 7
Jumlah Term pada Semua Dokumen

No	Term
1	<i>racist</i>
2	<i>chicago</i>
3	<i>people</i>
4	<i>cop</i>
5	<i>catch</i>
6	<i>beat</i>
7	<i>black</i>
8	<i>alabama</i>
9	<i>area</i>
10	<i>thief</i>
11	<i>local</i>
12	<i>police</i>

15. Menghitung Bobot Term Pada Masing-Masing Dokumen

Tahap selanjutnya mencari representasi nilai dari tiap dokumen dalam dataset. Dari sini akan dibentuk suatu vektor dari setiap term dalam dokumen. Sebagai contoh kita hitung bobot (*w*) dari *term cop* dalam Dokumen 1, seperti berikut :

Jumlah kemunculan term *cop* dalam Dokumen 1 (D1) sebanyak dua kali maka nilai (*tf* = 2), total keseluruhan dokumen sebanyak tiga maka nilai (*D*)=3. Kemudian dari ketiga dokumen tersebut, kemunculan term *cop* hanya berada pada Dokumen 1 (D1) saja, maka nilai (*df*) = 1 karena jumlah dokumen yang mengandung term *cop* hanya satu dokumen, sehingga dapat diperoleh nilai bobot dari *term cop* pada Dokumen 1 (D1) ialah 0,9542, dengan cara dihitung seperti berikut :

$$w_{ij} = tf_{ij} \times \log\left(\frac{D}{df_j}\right)$$

$$w_{ij} = 2 \times \log\left(\frac{3}{1}\right)$$

$$w_{ij} = 2 \times 0.477$$

$$w_{ij} = 0.9542$$

Kemudian perhitungan tersebut diterapkan pada setiap term yang ada pada dokumen, dengan demikian dapat diperoleh nilai bobot (*w*) untuk setiap term dalam masing-masing dokumen, ditunjukkan pada Tabel 9.

Tabel 8
Bobot Term Dari Masing-Masing Dokumen

term	tf			df	D/df	idf = (Log D/df)	w = tf*idf		
	D1	D2	D3				D1	D2	D3
<i>racist</i>	1	0	0	1	3	0,477121	0,477121	0	0
<i>chicago</i>	1	0	0	1	3	0,477121	0,477121	0	0
<i>people</i>	0	1	0	1	3	0,477121	0	0,477121	0
<i>cop</i>	2	0	0	1	3	0,477121	0,95424	0	0

<i>catch</i>	0	1	1	2	1,5	0,176091	0	0,176091	0,176091
<i>beat</i>	1	0	0	1	3	0,477121	0,477121	0	0
<i>black</i>	1	0	0	1	3	0,477121	0,477121	0	0
<i>alabama</i>	0	1	0	1	3	0,477121	0	0,477121	0
<i>area</i>	0	1	0	1	3	0,477121	0	0,477121	0
<i>thief</i>	0	1	1	2	1,5	0,176091	0	0,176091	0,176091
<i>local</i>	0	0	1	1	3	0,477121	0	0	0,477121
<i>police</i>	0	0	1	1	3	0,477121	0	0	0,477121
Nilai bobot setiap Dokumen =							2,862727	1,783545	1,306424

16. Klasifikasi dengan *Bernoulli Naïve Bayes*

Hasil dari ekstraksi fitur akan digunakan untuk proses klasifikasi. *Classifier* yang digunakan dalam penelitian ini adalah *Bernoulli Naïve Bayes*. Dasar dari pengklasifikasi *Naive Bayes* adalah Teorema Bayes. Pengklasifikasi *Naive Bayes* bekerja dengan prinsip bahwa semua fitur rahasia adalah independen satu sama lain. Secara matematis, teorema Bayes dinyatakan sebagai berikut.

$$P(A|B) = P(A) \cdot \frac{P(B|A)}{P(B)}$$

A dan B adalah dua peristiwa yang independen. $P(A | B)$ adalah probabilitas satu peristiwa ketika peristiwa lain telah terjadi. $P(A)$ dan $P(B)$ adalah probabilitas dari dua peristiwa independen A dan B. $P(B | A)$ adalah probabilitas peristiwa B sehubungan dengan peristiwa A. *Bernoulli Naïve Bayes* memiliki basis model yaitu probabilitas bersyarat dan dihitung dengan menggunakan :

$$P(f|w) = \frac{P(w|f) \cdot P(f)}{(P(w|f) \cdot P(f) + P(w|r) \cdot P(r))}$$

$P(f | w)$ adalah probabilitas sebuah artikel berita palsu jika mengandung kata-kata tertentu seperti yang disebutkan dalam dataset. $P(w | f)$ adalah probabilitas kata-kata ditemukan dalam artikel palsu. $P(f)$ adalah kemungkinan keseluruhan artikel berita palsu. Demikian pula, $P(w | r)$ adalah probabilitas kemunculan kata dalam artikel asli. Dan $P(r)$ adalah probabilitas keseluruhan dari artikel berita asli.

17. Evaluasi Model *Bernoulli Naïve Bayes*

Model *Bernoulli Naïve Bayes* dibentuk dengan data latih sebanyak 8800 data berita, menghasilkan nilai rata-rata akurasi sebesar 98,75%. Hasil tersebut didapatkan berdasarkan metode *K-Fold Cross Validation* dengan nilai k sebanyak 10 iterasi yang diterapkan pada proses pembentukan model.

Model *Bernoulli Naïve Bayes* yang telah dibentuk akan diujikan pada data uji (*news_test*, dan *label_test*) dengan jumlah data uji sebanyak 2200 data atau 20% dari total data dalam dataset. Pengujian dilakukan dengan cara menyesuaikan hasil prediksi model BNB pada data uji (*news_test*) terhadap *actual value*/nilai sebenarnya dari label data uji (*label_test*).

18. Pengujian Performa Model terhadap Keseluruhan Data Uji

Pengujian ini dilakukan berdasarkan nilai *confusion matrix* yang didapatkan, ditunjukkan pada Tabel 10.

Tabel 9
Hasil Confusion Matrix

<i>True Positive (TP)</i>	<i>False Positive (FP)</i>	<i>True Negative (TN)</i>	<i>False Negative (FN)</i>
1086	9	1081	24

Hasil dari *confusion matrix* ini kemudian dihitung dengan beberapa rumus, yaitu :

$$accuracy = \frac{(TP+TN)}{TP+FP+TN+FN}$$

$$accuracy = \frac{(1086+1081)}{1086+9+1081+24} = 0.985 \times 100 = 98,5\%$$

$$precision = \frac{TP}{TP+FP}$$

$$precision = \frac{1086}{(1086+9)} = 0.991 \times 100 = 99,1\%$$

$$recall = \frac{(TP)}{TP+FN}$$

$$recall = \frac{(1086)}{1086+24} = 0.978 \times 100 = 97,8\%$$

$$f1_score = \frac{2 \times precision \times recall}{(precision+recall)}$$

$$f1_score = \frac{2 \times 0.991 \times 0.978}{(0.991+0.978)} = 0.984 \times 100 = 98,4\%$$

Berdasarkan perhitungan yang dilakukan dengan rumus diatas, dapat disimpulkan performa yang diperoleh model BNB dari keseluruhan data uji yaitu akurasi 98,5%, *precision* 99,1%, *recall* 97,8%, dan *f1-score* 98,4%.

19. Pengujian Akurasi Setiap Label

Pengujian ini dilakukan berdasarkan nilai confusion matrix yang didapatkan pada tabel 4.1. Pengujian akurasi ini berfokus pada persentase akurasi dari tiap label data uji, yang dihitung dengan rumus berikut :

$$Fake = \frac{(FP)}{(Jumlah\ Label\ Fake)}$$

$$Fake = \frac{(9)}{(1110)} = 0.978 \times 100 = 97,8\%$$

$$True = \frac{(TN)}{(Jumlah\ Label\ True)}$$

$$True = \frac{(1081)}{(1090)} = 0.991 \times 100 = 99,1$$

Hasil dari perhitungan akurasi yang dilakukan dapat dilihat pada Tabel 4 berikut ini.

Tabel 10
Akurasi dari Setiap Label

No	Label Data Uji	Jumlah Data	Akurasi
1	Fake (1)	1110	97,8%
2	True (0)	1090	99,1%

Kesimpulan

Berdasarkan penelitian dan implementasi yang telah dilakukan penggunaan algoritma *Bernoulli Naïve Bayes* yang dikombinasikan dengan ekstraksi fitur TF-IDF mampu mengklasifikasikan berita hoax dengan baik. Model prediksi yang dibangun dengan 8800 data berita, mampu menghasilkan nilai akurasi sebesar 98,5% dari jumlah data uji sebanyak 2.200 data berita, dimana akurasi dari prediksi model untuk label 1 (*Fake*) sebesar 97,8%, dan akurasi untuk label 0 (*True*) sebesar 99,1%. Nilai *precision* menunjukkan berapa persen berita yang benar *Fake* dari keseluruhan berita yang diprediksi *Fake*, nilai *precision* yang didapatkan sebesar 99,1%. Nilai *recall* menunjukkan berapa persen berita yang diprediksi *Fake* dibandingkan keseluruhan berita yang sebenarnya *Fake*, nilai *recall* yang didapatkan sebesar 97,8%, dan terakhir nilai *f1-score* sebesar 98,4% yang menunjukkan perbandingan nilai rata-rata *precision* dan *recall* yang dibobotkan. Dari implementasi tersebut penggunaan ekstraksi fitur TF-IDF pada algoritma *Bernoulli Naive Bayes* mampu meningkatkan akurasi sebesar 16,08%, *precision* 15,7%, *recall* 16,22%, dan *f1-score* 15,92% bila dibandingkan dengan hasil penelitian sebelumnya (M. Singh et al., 2020).

Saran dari penelitian ini diharapkan untuk mengoptimalkan kombinasi dari algoritma dan ekstraksi fitur yang digunakan, dengan menambahkan deteksi berita hoax untuk bahasa yang berbeda serta dataset yang berbeda. Kemudian dari sistem yang dibangun sebagai pendukung penelitian ini, diharapkan dalam penelitian lain dapat menambahkan fungsi untuk melakukan *integration*, *reduction*, dan *cleaning* data karena perangkat lunak yang dibangun pada penelitian ini belum mampu menangani proses tersebut.

BIBLIOGRAFI

- Agarwal, V., Sultana, H. P., Malhotra, S., & Sarkar, A. (2019). *Analysis of Classifiers for Fake News Detection*. *Procedia Computer Science*, 165 (2019), 377–383. <https://doi.org/10.1016/j.procs.2020.01.035>. [Google Scholar](#)
- Ahmed, H., Traore, I., & Saad, S. (2018). *Detecting opinion spams and fake news using text classification*. *Security and Privacy*, 1 (1), e9. <https://doi.org/10.1002/spy2.9>. [Google Scholar](#)
- Ajao, O., Bhowmik, D., & Zargari, S. (2018). Fake news identification on Twitter with hybrid CNN and RNN models. *ACM International Conference Proceeding Series*, 226–230. <https://doi.org/10.1145/3217804.3217917>. [Google Scholar](#)
- Bondielli, A., & Marcelloni, F. (2019). A survey on fake news and rumour detection techniques. *Information Sciences*, 497, 38–55. <https://doi.org/10.1016/j.ins.2019.05.035>. [Google Scholar](#)
- Budiman, A., Young, J. C., & Suryadibrata, A. (2021). Implementasi Algoritma Naïve Bayes untuk Klasifikasi Konten Twitter dengan Indikasi Depresi. *Jurnal Informatika: Jurnal Pengembangan IT*, 6(2), 133–138. [Google Scholar](#)
- Granik, M., & Mesyura, V. (2017). *Fake News Detection Using Naive Bayes Classifier*. 900–903. [Google Scholar](#)
- Gravanis, G., Vakali, A., Diamantaras, K., & Karadais, P. (2019). Behind the cues: A benchmarking study for fake news detection. *Expert Systems with Applications*, 128, 201–213. <https://doi.org/10.1016/j.eswa.2019.03.036>. [Google Scholar](#)
- Mulyani, E., Muhamad, F. P. B., & Cahyanto, K. A. (2021). Pengaruh N-Gram terhadap Klasifikasi Buku menggunakan Ekstraksi dan Seleksi Fitur pada Multinomial Naïve Bayes. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 5(1), 264–272. [Google Scholar](#)
- Poddar, K., Amali, G. B. D., & Umadevi, K. S. (2019). Comparison of Various Machine Learning Models for Accurate Detection of Fake News. *2019 Innovations in Power and Advanced Computing Technologies, i-PACT 2019*, 1–5. <https://doi.org/10.1109/i-PACT44901.2019.8960044>. [Google Scholar](#)
- Prasetijo, A. B., Isnanto, R. R., Eridani, D., Soetrisno, Y. A. A., Arfan, M., & Sofwan, A. (2017). Hoax detection system on Indonesian news sites based on text classification using SVM and SGD. *Proceedings - 2017 4th International Conference on Information Technology, Computer, and Electrical Engineering, ICITACEE 2017, 2018-Janua*, 45–49. <https://doi.org/10.1109/ICITACEE.2017.8257673>. [Google Scholar](#)
- Prasetyo, A., Septianto, B. D., Shidik, G. F., & Fanani, A. Z. (2019). Evaluation of feature extraction TF-IDF in Indonesian hoax news classification. *Proceedings -*

2019 *International Seminar on Application for Technology of Information and Communication: Industry 4.0: Retrospect, Prospect, and Challenges, ISemantic 2019*, 1–6. <https://doi.org/10.1109/ISEMANTIC.2019.8884291>. [Google Scholar](#)

Rahutomo, F., Pratiwi, I. Y. R., & Ramadhani, D. M. (2019). Eksperimen Naïve Bayes Pada Deteksi Berita Hoax Berbahasa Indonesia. *Jurnal Penelitian Komunikasi Dan Opini Publik*, 23(1). <https://doi.org/10.33299/jpkop.23.1.1805>. [Google Scholar](#)

Singh, G., Kumar, B., Gaur, L., & Tyagi, A. (2019). Comparison between Multinomial and Bernoulli Naïve Bayes for Text Classification. *2019 International Conference on Automation, Computational and Technology Management, ICACTM 2019*, 593–596. <https://doi.org/10.1109/ICACTM.2019.8776800>. [Google Scholar](#)

Singh, M., Wasim, M., Singh, H., & Mishra, U. (2020). Materials Today : Proceedings Performance of bernoulli ' s naive bayes classifier in the detection of fake news. *Materials Today: Proceedings*, xxxx. <https://doi.org/10.1016/j.matpr.2020.10.896>. [Google Scholar](#)

Wongso, R., Luwinda, F. A., Trisnajaya, B. C., Rusli, O., Wongso, R., Luwinda, F. A., Trisnajaya, B. C., & Rusli, O. (2017). ScienceDirect ScienceDirect News Article Text Classification in Indonesian Language News Article Text Classification in Indonesian Language. *Procedia Computer Science*, 116, 137–143. <https://doi.org/10.1016/j.procs.2017.10.039>. [Google Scholar](#)

Copyright holder:

Agri Yodi Prayoga, Asep Id Hadiana, Fajri Rakhmat Umbara (2021)

First publication right:

Jurnal Syntax Admiration

This article is licensed under:

