ISSN 2085-4552

# Implementasi Algoritma Complement dan Multinomial Naïve Bayes Classifier Pada Klasifikasi Kategori Berita Media Online

Muhammad Naufal Randhika[1], Julio Christian Young[2], Alethea Suryadibrata[3], Hadian Mandala [4]
[1,2,3,]Department of Informatics, Universitas Multimedia Nusantara, Tangerang, Indonesia
[4]Faculty of Engineering, Universitas Hamzanwadi, Nusa Tenggara Barat, Indonesia
[1]muhammad.randhika@student.umn.ac.id, [2]julio.christian@umn.ac.id, [3]alethea@umn.ac.id,
[4]hadian_mandala@hamzanwadi.ac.id

Diterima 13 Januari 2021
Disetujui 23 Mei 2021

***Abstract*— The development of computer technology and the dissemination of information via internet are increasing significantly from time to time. News is one of the information media which also increased. Conventional printed media has now been replaced by electronic media known as online news portals/ media. PT Merah Putih Media is one of developing online news media. There are three main categories (Lifestyles, Sports, and Indonesia) in its portal that still categorized manually by the editor in chief within the company. This study tried to test the suitability of two classification algorithms that able to replace the manual process, namely Multinomial Naïve Bayes (MNBC) and Complement Naïve Bayes (CNBC). Moreover, experiments related to the combination of both of algorithms was also tried. Based on series of experiments which had conducted, we found that the combination of MNBC and CNBC models are able to achieve the F1-Score of 90.13%.**

***Keywords*— PT Merah Putih Media, Text Mining, Multinomial Naïve Bayes Classifier, Complement Naïve Bayes Classifier**

## I. LATAR BELAKANG

Perkembangan teknologi dan penyebaran informasi di internet selalu meningkat dari waktu ke waktu. Berita merupakan salah satu media informasi yang juga turut mengalami peningkatan [1].

PT. Merah Putih Media merupakan media berita online. Media berita online milik PT. Merah Putih dapat diakses pada alamat URL merahputih.com. Berita yang disampaikan terdiri dari tiga kategori mulai dari berita tentang Indonesia, Hiburan dan Gaya Hidup, serta Olahraga. Namun, pembagian artikel berita ke dalam kategori dilakukan secara manual oleh kepala redaksi jurnalis. Padahal hal ini bukan bagian dari deskripsi pekerjaannya dan hingga saat ini belum ada yang melamar pekerjaan operator kategori berita. Hal ini sering merepotkan ketika jumlah berita yang ingin diterbitkan banyak. Maka dari itu dibutuhkan sebuah sistem untuk mengklasifikasi berita [2].

Penelitian terhadap klasifikasi kategori berita merahputih.com milik PT Merah Putih Media akan menggunakan teknik Text Classification dengan algoritma Multinomial Naïve Bayes Classifier dikombinasikan dengan algoritma Complement Naïve Bayes Classifier dan menggunakan ekstraksi fitur TF-IDF.

Penelitian sebelumnya yang berkaitan dengan klasifikasi berita menggunakan algoritma Multinomial Naïve Bayes Classifier pernah dilakukan oleh Rahman, Wiranto, dan Afrizal, di mana penelitiannya mencoba mengklasifikasikan berita yang diperoleh berdasar lima media *online* ternama yaitu Detik, Viva, inilah.com, AntaraNews dan Okezone. Penelitian ini mencoba mengkategorikan berita-berita tersebut ke dalam tiga kategori besar yaitu kategori Kesejahteraan Rakyat, Ekonomi, dan Politik, Hukum dan Keamanan (Polhukam). Hasil penelitian sebelumnya menunjukkan bahwa metode Multinomial Naïve Bayes Classifier dan metode ekstraksi fitur TF-IDF menghasilkan nilai akurasi akhir sebesar 94,29% [3].

Penelitian lain yang berkaitan dengan klasifikasi berita dilakukan oleh Siti Nur Asiyah di mana penelitiannya membandingkan metode Support Vector Machine (SVM) dan K-Nearest Neighbor (KNN) untuk mengklasifikasi berita online. Penelitian ini menggunakan data yang berasal dari situs berita online detik.com. Penelitian ini mencoba mengklasifikasikan berita-berita ke dalam lima kategori yang terdiri dari news, finance, hot, sport, dan otomotif. Hasil dari penelitian menunjukkan bahwa metode SVM dapat menghasilkan nilai akurasi sebesar 93,14% dan metode K-NN menghasilkan nilai akurasi sebesar 68,90% [4].

Penelitian yang serupa tentang klasifikasi berita Online juga dilakukan oleh Bening, Dian, dan Lailil, di mana penelitiannya bertujuan untuk menghindari *human error* ketika akan menerbitkan berita di kategori yang tidak tepat pada portal berita online. Penelitian ini mencoba menguji tingkat performa dari metode pembobotan TF-IDF dan metode Cosine Similarity. Hasil dari penelitian menunjukkan bahwa gabungan

dari kedua metode mampu memperoleh nilai akurasi sebesar 91,25% [5].

Berdasarkan kajian penelitian di atas, dapat dilihat bahwa algoritma Multinomial Naïve Bayes memiliki akurasi yang paling baik. Maka penelitian ini juga menggunakan algoritma Multinomial Naïve Bayes dan untuk meningkatkan akurasi klasifikasi, penelitian ini akan mengkombinasikan dengan algoritma Complement Naïve bayes. Menurut [6] bahwa algoritma ini dapat memvalidasi prediksi klasifikasi dari algoritma Multinomial Naïve Bayes.

## II. STUDI PUSTAKA

### A. *Text Mining*

*Text Mining* adalah serangkaian metode komputasi yang memiliki kemampuan untuk menggali informasi dalam kumpulan data teks dan menjalankan tugas klasifikasi untuk sebuah dokumen masukan secara otomatis. Sebagai sebuah bidang ilmu, *Text Mining* merupakan sub-bidang dari keilmuwan *Data Mining* yang memiliki kapabilitas untuk mengekstraksi pola tersembunyi dari sekumpulan data tekstual dalam jumlah yang masif. Istilah *Text Classification* merupakan salah satu bentuk pendekatan Text Mining yang dapat digunakan untuk mengembangkan sebuah model klasifikasi teks [7].

*Text Classification* merupakan pemberian label dengan kategori yang sudah ditentukan pada sebuah teks dengan bahasa alami [8]. Klasifikasi teks diterapkan dalam berbagai konteks, mulai dari pengindeksan sebuah dokumen berdasarkan kosa kata, pemfilteran sebuah dokumen, pembuatan metadata otomatis, dan pada macam aplikasi lainnya [8]. Ada beberapa cara umum dalam klasifikasi teks secara otomatis, yaitu *pre-processing*, *feature extraction/selection*, *modeling* menggunakan teknik pembelajaran mesin, serta *training* dan *testing* pada *classifier* [9].

*Text Processing* merupakan salah satu tahapan dalam *Text Mining*. *Text Processing* diimplementasikan untuk mentransformasikan data tekstual yang tidak terstruktur sehingga menjadi lebih terstruktur sebelum dilakukan tugas-tugas *Text Mining* lainnya[10].

### B. *Text Preprocessing*

Tahap *Text Preprocessing* tersusun atas tahap-tahap lain yang lebih kecil, yaitu *Case Folding*, Tokenisasi, *Filtering*, dan *Stemming*. Tahap-tahap ini bersifat opsional dan bergantung pada kebutuhan analisis data yang akan dilakukan [11].

*Case Folding* merupakan salah satu bentuk teknik *text preprocessing*. Tujuan dari proses ini adalah untuk menormalisasikan data dengan mengubah seluruh huruf yang muncul dalam kumpulan dokumen menjadi huruf kecil [3]. Pada proses ini juga dilakukan tahapan untuk menghilangkan karakter-karater yang dianggap tidak relevan dalam sebuah tugas analisis teks seperti: penghilangan tanda baca, angka dan karakter lain selain huruf alfabet [11]. Kemudian, pada tahap ini juga akan dilakukan penghapusan jumlah spasi yang berlebihan di awal dan akhir sebuah kata, teknik ini biasa disebut *whitespace removal* [11].

Tokenisasi merupakan proses pencacahan sebuah *string* masukan menjadi unit-unit terkecil (kata) yang menyusunnya [11]. Pada prinsipnya, tujuan dari dilakukan proses ini adalah untuk mengetahui unit-unit terkecil yang menyusun sebuah dokumen [12].

*Filtering* merupakan proses pemilihan kata-kata yang dianggap relevan untuk sebuah tugas analisis teks dari hasil tokenisasi. Proses *filtering* juga dikenal dengan istilah lain yakni, *stopword removal*. Dalam proses *stopword removal*, setiap kata yang muncul dalam dokumen akan diperiksa apakah kata tersebut menjadi bagian dari *stop list* ataupun *word list*. *Stop list* merujuk pada kumpulan kata yang dianggap tidak penting dan tidak memiliki makna tertentu untuk sebuah tugas analisis, sedangkan *word list* merujuk pada kumpulan kata yang memiliki definisi sebaliknya. Saat sebuah kata tergabung ke dalam *stop list*, kata tersebut akan dihilangkan dari teks aslinya [11].

*Stemming* merupakan proses yang dijalankan untuk mengembalikan sebuah kata menjadi kata dasar atau dengan kata lain proses menemukan kembali akar kata dari setiap kata yang menyusun sebuah dokumen terfilter. Setelah proses *stemming* selesai dijalankan, setiap kata yang berimbuhan akan kembali ke bentuk kata dasarnya. Proses *stemming* bertujuan untuk melakukan normalisasi terhadap data dan menghindari representasi teks dengan distribusi kata yang tersebar (*sparse distributed representations*) [11].

### C. *TF-IDF*

Metode TF-IDF adalah salah satu metode ekstraksi fitur yang bekerja dengan merepresentasikan sebuah dokumen menjadi sekumpulan bobot tertentu. Dalam metode TF-IDF, sekumpulan bobot yang merepresentasikan sebuah dokumen dihitung berdasarkan kemunculan setiap kata dalam sebuah dokumen yang telah dinormalisasikan dengan frekuensi kemunculan kata tersebut pada kumpulan dokumen. Proses normalisasi pada rumus perhitungan bobot TF-IDF bertujuan untuk mengetahui derajat kepentingan dari setiap kata yang menyusun sebuah dokumen. Berdasarkan penjelasan yang telah diberikan, rumus pembobotan tiap kata (term – *t*) dalam sebuah dokumen *d* dapat dilihat pada bagian di bawah ini [13],

$$w_{t,d} = tf_{t,d} \times \log\left(\frac{N}{df_t}\right) \quad (1)$$

dengan,

$w_{t,d}$ = Bobot TF-IDF
$tf_{t,d}$ = Banyaknya kata *t* dalam sebuah dokumen *d*
$N$ = Frekuensi dokumen

$df_t$ = Frekuensi dokumen yang mengandung kata *t*

*D. Multinomial dan Gaussian Naïve Bayes Classifier*

Multinomial Naïve Bayes Classifier (MNBC) merupakan model yang dibentuk berdasarkan Teorema Bayes yang terbukti memiliki performa yang baik untuk tugas klasifikasi teks. Dalam persamaan MNBC, tidak seperti persamaan pada algoritma Bernoulli Naïve Bayes yang hanya memperhitungkan jumlah kemunculan dokumen dengan kata tertentu dalam suatu kelas, frekuensi kemunculan kata dalam suatu kelas juga turut diperhitungkan [14]. Definisi formal dari MNBC untuk mengklasifikasikan kelas *c* untuk sebuah dokumen *d* dapat dilihat seperti pada bagian berikut,

$$MNBC(d) = argmax_{c \in C} P(c) \times \prod_{i<k<n_d} P(x_k|c) \quad (2)$$

dengan,

$P(c)$ = Probabilitas *prior* dari kelas (target)
$P(x_k|n)$ = Probabilitas kata ke-k dalam kelas *c* dibanding seluruh kelas lainnya.
$C$ = Himpunan seluruh kelas yang dapat dimiliki *d*
$n_d$ = Jumlah seluruh kata unik dalam *d*.

Akan tetapi, saat diimplementasikan, persamaan Multinomial Naïve Bayes Classifier (Persamaan 2) dapat menyebabkan kondisi *floating point underflow*, sehingga seringkali dilakukan tahap log normalization pada formula ini sehingga persamaannya berubah menjadi [15],

$$MNBC(d) = argmax_{c \in C} \left[ \log P(c) + \sum_{1<k<n_d} \log P(x_k|c) \right] \quad (3)$$

Di sisi lain, algoritma Complement Naive Bayes Classifier (CNBC) merupakan algoritma yang akan melengkapi Algoritma MNBC. Pada algoritma CNBC proses pengklasifikasian kelas *c* untuk sebuah dokumen *d* didefinisikan oleh persamaan di bawah ini [6],

$$CNBC(d) = argmax_{c \in C} \left[ \log P(c) + \sum_{1<k<n_d} \log P(x_k|c') \right] \quad (4)$$

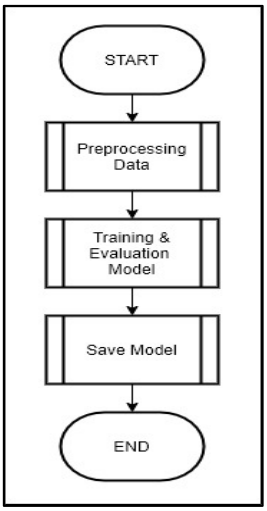
dengan nilai $P(x_k|c')$ merepresentasikan probabilitas kata ke-*k* yang diketahui pada kelas bukan *c*.

## III. METODOLOGI DAN PENELITIAN

Dalam penelitian yang akan dilakukan, bagian Flowchart Umum akan digunakan untuk menjelaskan tahapan-tahapan yang akan dilakukan untuk mengetahui tingkat performa dari model gabungan algoritma MNBC dan CNBC. Kemudian bagian-bagian lain di bawahnya akan digunakan untuk menjelaskan detail dari setiap tahap dalam Flowchart Umum.

*A. Flowchart Umum*

Dalam penelitian yang akan dilakukan detil dari proses-proses yang akan dilakukan ditunjukan oleh gambar berikut.

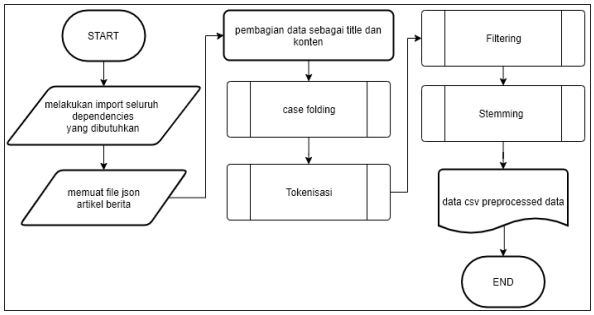



Gambar 1 Flowchart Umum

Sesuai dengan Gambar 1, tahap pertama yang dilakukan adalah *Preprocessing Data*. Kemudian, hasil dari proses akan digunakan untuk melakukan proses Model Training dan Evaluation. Pada proses ini akan dilakukan proses pelatihan model gabungan Algoritma Complement dan Multinomial Naïve Bayes Classifier menggunakan subset dari data yang tersedia. Setelah itu, berdasarkan model yang telah dilatih, proses evaluasi akan dilakukan dengan memberikan subset data lainnya yang belum pernah dikenali oleh model. Proses pelatihan dan evaluasi dari model akan dilakukan dengan mengujicobakan beberapa parameter dengan strategi *k-fold cross validation* dengan nilai *k* sama dengan 5. Terakhir, berdasarkan model dengan parameter terbaik, proses Save Model untuk menyimpan model data latih dan data uji yang terbaik.

*B. Preprocessing Data*

Dalam penelitian, data yang akan diolah terdiri dari 42.155 artikel dengan artikel terkait dengan Hiburan & Gaya Hidup sebesar 27.019, Olahraga sebesar 7.072, dan Indonesiaku sebesar 8.064. Pada proses Preprocessing Data, untuk setiap artikel, akan dilakukan tahap-tahap transformasi data tekstual seperti yang sudah dijelaskan pada bagian sebelumnya mengenai tahap-tahap Text Preprocessing. Urutan dari tahap-tahap Text Preprocessing yang akan dilakukan dalam penelitian digambarkan pada *flowchart* di Gambar 2.




Gambar 2 Flowchart Preprocessing Data

### C. *Training & Evaluation*

Proses Training & Evaluation dimulai dengan memuat data hasil Text Preprocessing beserta label yang dimiliki oleh setiap data. Dalam proses Training & Evaluation, tahap yang pertama kali dilakukan adalah inisialisasi model klasifikasi dan TF-IDF. Dalam eksperimen yang dilakukan, terdapat 3 buah model yang akan dilatih dan dievaluasi yaitu model MNBC, model CNBC, serta model gabungan yang terdiri dari metode MNBC dan CNBC. Untuk setiap model akan dilakukan tahap pengujian parameter berikut.

1. Teks yang akan digunakan sebagai masukan dari model (analyser). Dalam penelitian yang ini, analyser yang dicobakan terdiri dari judul, konten serta gabungan dari judul dan konten.
2. Parameter model yang akan mempengaruhi bagaimana fitur-fitur akan dibentuk oleh metode TF-IDF. Dalam penelitian akan diujikan analyser berupa *word* dan *character*. Saat analyser di-*set* nilainya menjadi *word* maka metode TF-IDF akan membentuk representasi fitur dari dokumen berdasarkan kata-kata yang terdapat dalam sebuah dokumen. Di sisi lain, saat analyser di-*set* nilainya menjadi *character* maka fitur akan dibentuk berdasarkan huruf-huruf yang menyusun setiap kata dalam tiap dokumen.
3. Parameter min DF dan max DF yang akan memfilter jumlah kemunculan kata (*terms*) berdasarkan suatu *threshold* tertentu. Pada metode TF-IDF, fitur hanya akan dibentuk berdasarkan kata-kata yang jumlah kemunculannya berada di antara min DF dan max DF.
4. Parameter ngram_range yang mengatur rentang nilai dari kombinasi kata ataupun huruf yang akan dianggap sebagai fitur.

Berdasarkan setiap model yang telah dibentuk dari setiap kombinasi parameter yang telah diberikan, akan dilakukan proses evaluasi dengan menggunakan metrik F1-Score. Berdasarkan setiap parameter teks masukan yang diuji dan parameter model akan dicatat nilai parameter dari model dengan tingkat F1-Score terbaik. Visualisasi dari proses pelatihan dan evaluasi yang dilakukan dapat dilihat pada Gambar 3 pada halaman selanjutnya.

### D. *Save Model*

Pada tahap ini, model dengan nilai F1-Score tertinggi yang dihasilkan pada tahap Train & Evaluation akan disimpan agar dapat digunakan oleh pihak terkait saat dibutuhkan.

## IV. Hasil Implementasi

### A. *F1-Score*

Berdasarkan tahap-tahap metodologi penelitian yang telah dijelaskan pada bagian sebelumnya, Gambar 4 (pada halaman selanjutnya) menunjukkan model-model dengan F1-Score terbaik untuk setiap analyser dan model yang diujicobakan.

Berdasarkan Gambar 4, terlihat bahwa model terbaik hasil pelatihan adalah model gabungan CNBC & MNBC dengan parameter *character model*, analyser berupa content & title, Max_DF sebesar 0,9 dan Min_DF sebesar 0,025, dan nilai ngram_range sebesar (3,4) dengan F1-Score sebesar 90,13%.

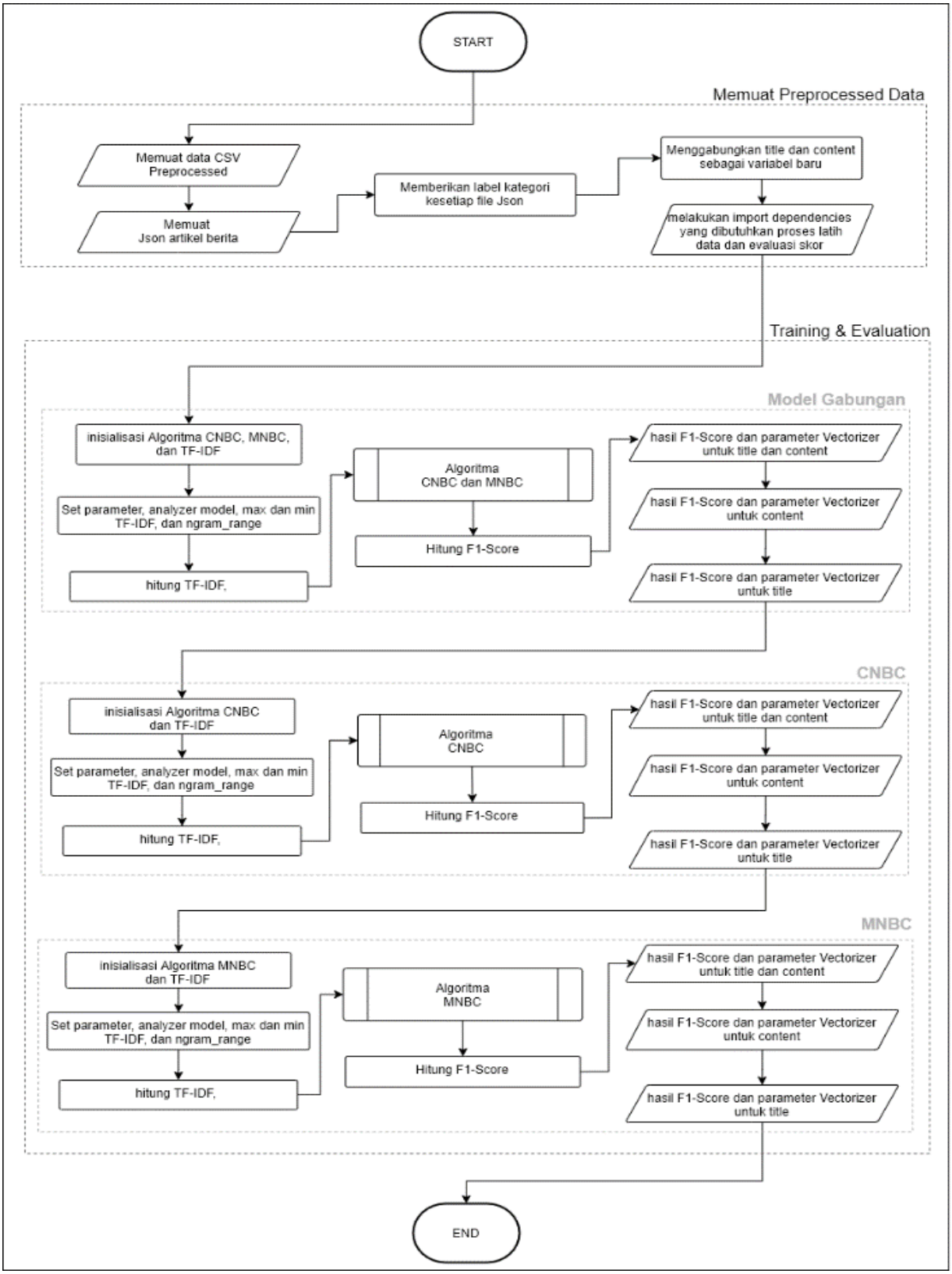
### B. *Confusion Matrix*

Berdasarkan model dengan performa F1-Score terbaik yang dihasilkan, dilakukan analisis lanjutan dengan menggunakan metrik pengukuran lainnya yaitu, *confusion matrix* untuk mengetahui ketepatan prediksi dari model untuk setiap kategori berita. Proses analisis lanjutan dilakukan dengan melatih model baru dengan parameter yang serupa dengan model terbaik. Model ini dilatih dengan menggunakan 80% dari data artikel yang ada dan akan dievaluasi menggunakan 20% data artikel lainnya yang tersisa. Hasil confusion matrix dari model dari model terbaik dapat dilihat seperti pada Tabel 1.

Tabel 1 Confusion Matrix CNBC dan MNBC

| | | SISTEM | | |
|---|---|---|---|---|
| | | Hiburan & Gaya Hidup | Olahraga | Indonesiaku |
| AKTUAL | Hiburan & Gaya Hidup | 4861 | 59 | 412 |
| | Olahraga | 55 | 1375 | 18 |
| | Indonesiaku | 330 | 8 | 1313 |

Melalui Tabel 1 dapat terlihat bahwa meskipun memiliki F1-Score yang sangat baik (90.13%), kesalahan prediksi paling kecil diperoleh oleh kategori Olahraga (sebesar 5% artikel dengan kategori Olahraga salah diprediksi) dan kesalahan prediksi paling besar diperoleh oleh kategori Indonesiaku (sebesar 20% artikel dengan kategori Indonesiaku salah diprediksi). Hal ini menunjukkan bahwa artikel dengan kategori Hiburan & Gaya Hidup dan Indonesiaku mungkin saja memiliki fitur yang serupa (cenderung tersusun berdasarkan kata-kata yang sama). Kemudian, berdasarkan confusion matrix yang dihasilkan, dihitung nilai precision, recall, serta F1-Score dari masing-masing kelas. Tabel 2 (halaman selanjutnya) menunjukkan nilai performa untuk masing-masing kategori.




Gambar 3 Flowchart Training & Evaluation Model

| No | Methods | Model | Analyzer | Max_DF | Min_DF | N_GRAM_RANGE | F1-Score |
|---|---|---|---|---|---|---|---|
| 1 | CNBC & MNBC | Word Model | Content & Title | 0,9 | 0,025 | (1, 3) | 89,28% |
| | | | Content | 0,9 | 0,025 | (1, 3) | 89,23% |
| | | | Title | 0,9 | 0,025 | (1, 1) | 63,37% |
| | | | | | | | |
| | | Char Model | Content & Title | 0,95 | 0,025 | (3, 4) | 90,13% |
| | | | Content | 0,9 | 0,025 | (3, 4) | 90,05% |
| | | | Title | 0,9 | 0,025 | (2, 3) | 71,27% |
| | | | | | | | |
| 2 | CNBC | Word Model | Content & Title | 0,9 | 0,025 | (1, 3) | 88,55% |
| | | | Content | 0,9 | 0,025 | (1, 3) | 88,55% |
| | | | Title | 0,9 | 0,05 | (1, 1) | 62,86% |
| | | | | | | | |
| | | Char Model | Content & Title | 0,9 | 0,025 | (3, 4) | 89,53% |
| | | | Content | 0,9 | 0,025 | (3, 4) | 89,45% |
| | | | Title | 0,9 | 0,025 | (2, 3) | 66,31% |
| | | | | | | | |
| 3 | MNBC | Word Model | Content & Title | 0,9 | 0,025 | (1, 3) | 89,15% |
| | | | Content | 0,9 | 0,025 | (1, 3) | 89,08% |
| | | | Title | 0,9 | 0,025 | (1, 1) | 62,99% |
| | | | | | | | |
| | | Char Model | Content & Title | 0,9 | 0,025 | (3, 4) | 90,03% |
| | | | Content | 0,95 | 0,025 | (3, 4) | 89,96% |
| | | | Title | 0,9 | 0,025 | (2, 3) | 68,71% |

Gambar 4 Tabel Evaluasi F1-Score

Tabel 2 Rangkuman Perhitungan Rasio

| Kategori | Precision | Recall | F1-Score |
|---|---|---|---|
| Hiburan & Gaya Hidup | 0.93 | 0.91 | 0.92 |
| Olahraga | 0.95 | 0.95 | 0.95 |
| Indonesiaku | 0.75 | 0.80 | 0.77 |

Melalui Tabel 2 dapat terlihat bahwa model terbaik yang dihasilkan cenderung memiliki performa F1-Score yang baik saat memprediksi kelas Hiburan & Gaya Hidup dan Olahraga. Di sisi lain, untuk kelas Indonesiaku, model cenderung memiliki performa F1-Score yang kurang memuaskan. Hal ini mungkin saja disebabkan oleh jumlah data dari kelas Indonesiaku yang berjumah sangat sedikit jika dibandingkan dengan jumlah data kelas lainnya (berbeda sampai dengan 992 artikel dengan kelas Olahraga dan berbeda hampir 20.000 data dengan kelas Hiburan & Gaya Hidup). Untuk meningkatkan performa dari kelas Indonesiaku tentunya dapat diujicobakan model-model lain yang bersifat lebih stabil untuk data yang bersifat *imbalance*.

## V. KESIMPULAN DAN SARAN

Berdasarkan hasil implementasi yang sudah dilakukan, maka penelitian dapat disimpulkan bahwa implementasi model algoritma Complement dan Multinomial Naïve Bayes Classifier dapat untuk mengklasifikasikan Kategori berita merahputih.com milik PT Merah Putih Media berhasil dilakukan. Berdasarkan percobaan yang telah dilakukan, dihasilkan model terbaik dengan performa F1-Score sebesar 90.13%.

Berdasarkan penelitian yang telah dilakukan, terdapat beberapa saran untuk pengembangan penelitian lanjutan, antara lain.

1. Mencari model algoritma yang memiliki tingkat kompleksitas lebih tinggi. Dataset yang digunakan dalam penelitian ini memiliki jumlah yang besar tetapi model gabungan yang digunakan dalam penelitian memiliki tingkat kompleksitas yang cukup sederhana, sehingga apabila memiliki jumlah dataset yang besar dan memiliki model algoritma dengan tingkat kompleksitas tinggi dapat meningkatkan akurasi. Apabila ingin melakukan pengembangan penelitian, maka model yang disarankan adalah XGBoost Classifier atau Deep Neural Network Classifier.
2. Melakukan Entity Recognition. Model gabungan yang digunakan memiliki tingkat performa yang akurat. Alangkah baiknya jika dalam satu kategori, diketahui seluruh named-entity yang muncul, hal ini dapat memudahkan dan membantu proses Search Engine Optimization (SEO).

## DAFTAR PUSTAKA

[1] Y. Wibisono and L. M. Khodra, "Clustering Berita Berbahasa Indonesia," *FMIPA UPI & STEI ITB*, p. 32, 2005.

[2] J. Irawan, Interviewee, *HRD PT Merah Putih Media.* [Interview]. 16 March 2020.

[3] R. Feldman and J. Sanger, The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data, New York: Cambridge University Press, 2007.

[4] A. Rahman, Wiranto and A. Doewes, "Online News Classification using Multinomial Naive Bayes," *ITSMART: Jurnal Ilmiah Teknologi dan Informasi*, p. 32, 2017.

[5] N. A. Siti, "KLASIFIKASI BERITA ONLINE MENGGUNAKAN METODE SUPPORT VECTOR MACHINE DAN K-NEAREST NEIGHBOR," INSTITUT TEKNOLOGI SEPULUH NOPEMBER, SURABAYA, 2016.

[6] B. Herwijayanti, E. D. Ratnawati and L. Muflikhah, "Klasifikasi Berita Online dengan menggunakan Pembobotan TF-IDF dan Cosine Similarity," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, pp. 306-312, 2018.

[7] D. M. J. Rennie, L. Shih, J. Teevan and R. D. Krager, "Tackling the Poor Assumptions of Naive Bayes Text Classifiers," *International Conference on Machine Learning*, pp. 1-8, 2003.

[8] F. Sebastiani, "Machine Learning in Automated Text Categorization: a Bibliography," *ACM Computing Surveys*, pp. 1-47, 2001.

[9] K. M. Dalal and A. M. Zaveri, "Automatic Text Classification: A Technical Review," *International Journal of Computer Applications*, pp. 37-40, 2011.

[10] P. D. Langgeni, A. Z. Baizal and F. Y. A.W., "Clustering Artikel Berita Berbahasa Indonesia Menggunakan Unsupervised Feature Selection," *Seminar Nasional Informatika UPN*, pp. D1-D10, 2010.

[11] S. M. Hudin, "IMPLEMENTASI METODE TEXT MINING DAN K-MEANS CLUSTERING UNTUK PENGELOMPOKAN DOKUMEN SKRIPSI (STUDI KASUS BRAWIJAYA)," UNIVERSITAS BRAWIJAYA, MALANG, 2018.

[12] J. Asian, E. H. Williams and M. M. S. Tahaghoghi, "A Testbed for Indonesian Text Retrieval," *Australasian Document Computing Symposium*, pp. 1-4, 2004.

[13] A. A. A. Putra, "Implementasi Text Summarization Menggunakan Metode Vector Space Model Pada Artikel Berita Berbahasa Indonesia," Perpustakaan UNIKOM, Bandung, 2016.

[14] F. Azuaje, I. Witten and F. E., Data Mining: Practical Machine Learning Tools and Techniques, Northen Ireland: BioMedical Engineering OnLine , 2006.

[15] D. C. Manning, P. Raghavan and H. Schutze, Introduction to Information Retrieval, Cambirdge: Cambridge University Press, 2008.

[16] P. U. D. Singh, A. Tiwari and K. R. Singh, Soft-Computing-Based Nonlinear Control Systems Design, Madhya Pradesh: Madhav Institute of Technology and Science, 2018.