Matrik: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer
Vol. 21, No. 3, Juli 2022, pp. 653∼666
ISSN: 2476-9843, accredited by Kemenristekdikti, Decree No: 200/M/KPT/2020
DOI: 10.30812/matrik.v21i3.1851

Komparasi Ekstraksi Fitur dalam Klasifikasi Teks Multilabel Menggunakan Algoritma *Machine Learning*

Comparison of Feature Extraction in Multilabel Text Classification Using Machine Learning Algorithm

Lusiana Efrizoni[1], Sarjon Defit[2], Muhammad Tajuddin[3], Anthony Anggrawan[4]
[1]STMIK Amik Riau, Indonesia
[2]Universitas Putra Indonesia YPTK Padang, Indonesia
[3,4]Universitas Bumigora, Indonesia

Informasi Artikel

Genesis Artikel:

Diterima, 08 April 2022
Direvisi, 11 Mei 2022
Disetujui, 06 Juli 2022

Kata Kunci:

Ektraksi Fitur
Klasifikasi Teks Multilabel
Machine Learning
Perbandingan Kinerja Model

ABSTRAK

Ektraksi fitur dan algoritma klasifikasi teks merupakan bagian penting dari pekerjaan klasifikasi teks, yang memiliki dampak langsung pada efek klasifikasi teks. Algoritma *machine learning* tradisional seperti *Naïve Bayes*, *Support Vector Machines*, *Decision Tree*, *K-Nearest Neighbors*, *Random Forest*, *Logistic Regression* telah berhasil dalam melakukan klasifikasi teks dengan ektraksi fitur i.e. *Bag of Word (BoW)*, *Term Frequency-Inverse Document Frequency (TF-IDF)*, *Documents to Vector (Doc2Vec)*, *Word to Vector (word2Vec)*. Namun, bagaimana menggunakan vektor kata untuk merepresentasikan teks pada klasifikasi teks menggunakan algoritma *machine learning* dengan lebih baik selalu menjadi poin yang sulit dalam pekerjaan *Natural Language Processing* saat ini. Makalah ini bertujuan untuk membandingkan kinerja dari ekstraksi fitur seperti BoW, TF-IDF, Doc2Vec dan Word2Vec dalam melakukan klasifikasi teks dengan menggunakan algoritma *machine learning*. Dataset yang digunakan sebanyak 1000 sample yang berasal dari tribunnews.com dengan split data 50:50, 70:30, 80:20 dan 90:10. Hasil dari percobaan menunjukkan bahwa algoritma *Naïve Bayes* memiliki akurasi tertinggi dengan menggunakan ekstraksi fitur TF-IDF sebesar 87% dan BoW sebesar 83%. Untuk ekstraksi fitur Doc2Vec, akurasi tertinggi pada algoritma SVM sebesar 81%. Sedangkan ekstraksi fitur Word2Vec dengan algoritma *machine learning* (i.e. i.e. *Naïve Bayes*, *Support Vector Machines*, *Decision Tree*, *K-Nearest Neighbors*, *Random Forest*, *Logistic Regression*) memiliki akurasi model dibawah 50%. Hal ini menyatakan, bahwa Word2Vec kurang optimal digunakan bersama algoritma *machine learning*, khususnya pada dataset tribunnews.com.

Keywords:

Feature Extraction
Machine Learning
Model Performance
Comparison
Text Classification
Multilabel

ABSTRACT

Feature extraction and text classification algorithms are an important part of text classification work, which has a direct impact on the text classification effect. Traditional machine learning algorithms such as Naïve Bayes, Support Vector Machines, Decision Tree, K-Nearest Neighbors, Random Forest, Logistic Regression have succeeded in classifying text with feature extraction i.e. Bag of Word (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), Documents to Vector (Doc2Vec), Word to Vector (word2Vec). However, how to use word vectors to represent text in text classification using machine learning algorithms better has always been a difficult point in Natural Language Processing work nowadays. This paper aims to compare the performance of feature extraction such as BoW, TF-IDF, Doc2Vec and Word2Vec in performing text classification using machine learning algorithms. The dataset used is 1000 samples from tribunnews.com with split data 50:50, 70:30, 80:20 and 90:10. The results of the experiment show that the Naïve Bayes algorithm has the highest accuracy using TF-IDF feature extraction of 87% and BoW of 83%. For Doc2Vec feature extraction, the highest accuracy in the SVM algorithm is 81%. Meanwhile, Word2Vec feature extraction with machine learning algorithms (i.e. i.e. Naïve Bayes, Support Vector Machines, Decision Tree, K-Nearest Neighbors, Random Forest, Logistic Regression) has a model accuracy below 50%. This means that Word2Vec is not optimal for use with machine learning algorithms, especially in the tribunnews.com dataset.

Penulis Korespondensi:

Lusiana Efrizoni,
Program Studi Teknik Informatika,
STMIK Amik Riau,Indonesia
Email: lusiana@stmik-amik-riau.ac.id.

1. PENDAHULUAN

Ekstraksi fitur (*Feature Extraction*) dikenal juga dengan istilah penyisipan kata (*Word Embedding*). Salah satu topik menarik dalam penelitian bidang Pemrosesan Bahasa Alami (*Natural Language Processing*) adalah *Feature Extraction* atau *Word Embedding* [1]. *Word embedding* mulai dikembangkan sekitar tahun 2000 [2, 3]. Cara kerja *word embedding* memetakan setiap kata dalam dokumen ke dalam *dense vector*, dimana sebuah *vector* merepresentasikan proyeksi kata di dalam ruang *vector* [4, 5]. Posisi kata tersebut dipelajari dari teks atau berdasarkan kata-kata disekitarnya. *Word embedding* dapat menangkap makna semantik dan sintaktik kata. *Word embedding* juga dapat digunakan untuk menghitung kesamaan kata, seperti *information retrieval* [6]. Dalam artikel ini, model *word embedding* yang diuji coba adalah *Bag of Word (BoW)*, *Term Frequency Inverse Document Frequency (TF-IDF)*, *Doc2Vec* dan *Word2Vec*.

Peningkatan data tekstual *online* telah menjadi hal yang sulit bagi pengguna untuk mengakses konten yang diminati sehingga perlu untuk mengklasifikasikan atau mengkategorikan teks agar mudah diakses [7]. Sering sekali ditemukan judul artikel berita dengan satu topik saja namun di dalam artikel tersebut bisa saja mengandung satu atau lebih topik berita. Berita atau informasi di media sosial, memungkinkan terjadinya kesalahan dalam melakukan pengelompokkan berita, seperti suatu berita dikategorikan pada kategori *infotainments*. Sedangkan berdasarkan isi berita atau kata- kata yang terkandung di dalamnya, berita tersebut seharusnya dikategorikan pada kategori politik. Klasifikasi teks otomatis dikembangkan karena pekerjaan manual tidak lagi efektif. Jika dilakukan secara otomatis, orang tidak akan diminta untuk berpikir tentang kategori mana teks itu berada [8]. Kemampuan untuk mengklasifikasikan teks (dokumen) ke dalam kategori tertentu sangat membantu untuk menghadapi informasi yang berlebihan [9].

Klasifikasi teks dalam *Natural Language Processing (NLP)* sudah diterapkan pada aplikasi seperti *indexing* [10], *ranking* [11], *sentiment analysis* [12], *retrieval information* [13], dan klasifikasi dokumen [14]. Pelabelan dokumen baru sesuai dengan kategori yang benar, tergantung pada banyaknya dokumen berlabel yang ada untuk referensi [15]. Penelitian klasifikasi teks menggunakan algoritma *machine learning* seperti *k-Nearest Neighbors* [15, 16], *Naïve Bayes* [17–19], *Support Vector Machine* [15, 20, 21], *Logistic Regression* [9] dan *K-Means* [22, 23] sudah banyak dilakukan oleh peneliti sebelumnya. *Naïve Bayes Classifiers (NBC)*, *Support Vector Machine (SVM)* dan *k-Nearest Neighbors (KNN)* merupakan *classifier* yang paling banyak digunakan untuk menyelesaikan masalah *Document Classification* dan keduanya memberikan hasil yang cukup menjanjikan. Penelitian yang dilakukan oleh Rini Wongso dkk. [7], menunjukkan bahwa kombinasi TF-IDF dan *Multinomial Naïve Bayes Classifier* memberikan hasil terbaik dibanding SVM dengan *precision* sebesar 0.9841519, *recall* sebesar 0.98400 dan *accuracy* 85%. Penelitian yang dilakukan oleh Azam dkk. [16] menujukkan kinerja KNN 7% lebih baik dibandingkan dengan NBC. Sementara, penelitian yang dilakukan oleh Gunawan [24] melakukan klasifikasi teks menggunakan SVM pada dokumen berita CNN Indonesia berdasarkan lima kategori (Politik, Ekonomi, Teknologi, Olahraga, dan Hiburan), *accuracy* yang diperoleh 90%. Dari penelitian sebelumnya, maksimal algoritma yang dibandingkan dalam melakukan klasifikasi teks atau dokumen adalah tiga algoritma dengan *feature extraction* maksimal dua. Sedangkan pada penelitian ini membandingkan enam algoritma *Machine Learning* dengan empat *feature extraction* (i.e BOW, TF-IDF, Doc2Vec dan Word2Vec).

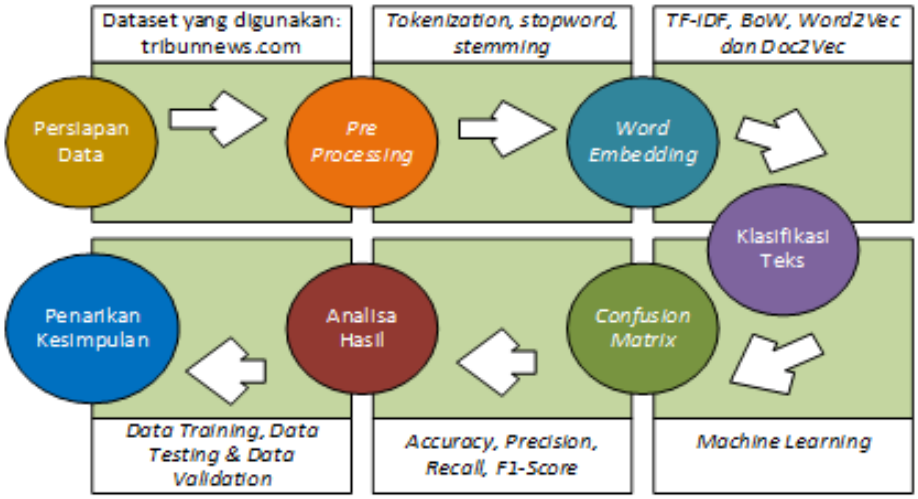
Pada penelitian sebelumnya, komparasi yang dilakukan maksimum pada dua algoritma *machine learning* yang dikombinasikan dengan satu atau dua model *feature extraction* atau *word embedding*. Sementara pada penelitian ini, melakukan komprasi dari enam algoritma *machine learning* yang digunakan pada klasifikasi teks multilabel seperti *Naïve Bayes*, *Support Vector Machines (SVM)*, *Decision Tree*, *K-Nearest Neighbors (KNN)*, *Random Forest*, *Logistic Regression* dikombinasikan dengan empat model *feature extraction* (i.e. BoW, TF-IDf, Doc2Vec dan Word2Vec). Enam algoritma *machine learning* ini merupakan algoritma yang popular digunakan dalam klasifikasi teks, meskipun masih memiliki keterbatasan dalam kasus pelatihan dataset skala besar [21]. Kombinasi *feature extraction* dan algoritma klasifikasi teks merupakan bagian dari pekerjaan klasifikasi teks, yang memiliki dampak langsung pada kalsifikasi teks. Namun, bagaimana menggunakan *vector* kata (*feature extraction*) untuk menyajikan teks lebih baik dalam algoritma *machine learning* tradisional, selalu menjadi poin yang sulit dalam NLP [25].

Penelitian ini bertujuan, untuk memilih algoritma *machine learning* yang paling sesuai dengan empat teknik fitur ekstraksi yang digunakan melalui proses komparasi. Hasil komparasi akan menentukan algoritma yang memiliki kinerja optimal dalam melakukan klasifikasi teks multilabel pada artikel berita bahasa Indonesia dengan beberapa split data yang berbeda, khususnya pada dataset tribunnews.com. Penelitian terkait komparasi telah dilakukan oleh beberapa peneliti sebelumnya seperti komparasi terhadap metode *word embedding* (i.e. LSA, Word2Vec and GloVe) dalam segmentasi *topic* pada bahasa Arabic dan English [1]; analisa komparasi dari model *word embedding* seperti *Continuous bag of words*, *Skip gram*, *Glove*(*Global Vectors for word representation*) *and Hellinger PCA (Principal Component Analysis)* untuk representasi yang efisien dan ekspresif pada kesamaan kontekstual [4]; perbandingan representasi berbasis kata dan berbasis konteks (i.e. Word2Vec dan GloVe) untuk klasifikasi masalah dalam informatika kesehatan [5]; survei terhadap algoritma klasifikasi teks [3]; dan penelitian yang membandingkan tiga algoritma *machine learning Naïve Bayes Classifier, K-Nearest Neighbor (KNN)* dan *Decision Tree* untuk menganalisis sentimen pada interaksi netizen dan pemerintah [26]. Selain itu, seleksi fitur dapat digunakan untuk mengklasifikasi aroma jenis kopi arabika Gayo [27].

Organisasi penulisan dari manuskrip ini adalah bagian ke dua tentang metodologi penelitian yang menjelaskan secara ringkas metode yang digunakan dalam penelitian. Pada bagian ke tiga menjelaskan hasil dan analisis dari percobaan yang dilakukan pada klasifikasi teks multilabel menggunakan enam algoritma *machine learning* dengan masing-masing empat fitur ekstraksi. Pada bagian ke tiga juga menyajikan hasil komparasi yang diperoleh dari percobaan yang dilakukan dengan beberapa split data yang berbeda. Kesimpulan dari penelitian disajikan pada bagian ke empat dari *manuscript* ini.

2. METODE PENELITIAN

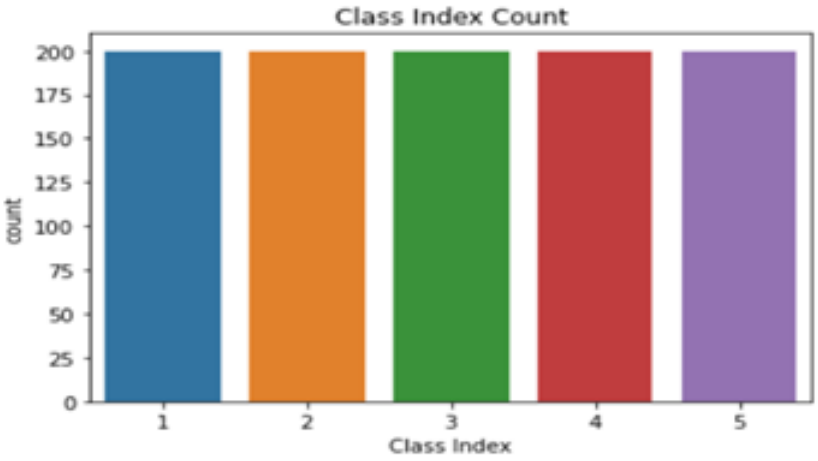
Secara garis besar, langkah-langkah dalam penelitian digambarkan dalam kerangka penelitian. Kerangka penelitian yang disajikan pada Gambar 1 terdiri dari persiapan data, *preprocessing*, *word embedding*, klasifikasi teks, *confusion matrix*, analisa hasil dan penarikan kesimpulan.



Gambar 1. Kerangka penelitian

2.1. Persiapan Data

Persiapan data merupakan tahap pengumpulan data. Dataset teks yang digunakan berasal dari tribunnews.com dengan lima kategori berita (i.e. *News*, *Business*, *Sport*, *Economy*, *Automotive*). Dataset dikumpulkan menggunakan *web scraping* dan disimpan dengan format *comma separated values* (.csv). Dataset dibagi menjadi data *training*, data *validation* dan data *testing* dengan banyaknya dataset 1000 *sample*. Dataset di-split ke dalam empat komposisi yaitu: 50:50 (50% data *training* dan 50% data *testing*), 70:30 (70% data *training* dan 30% data *testing*), 80:20 (80% data *training* dan 20% data *testing*), dan 90:10 (90% data *training* dan 10% data *testing*). Dataset telah dilabeli sebelumnya dengan salah satu dari 5 kategori, yaitu: *News (Class_index 1), Business (Class_Index 2), Sport (Class_Index 3), Economy (Class_Index 4) dan Automotive (Class_Index 5)*. Ke lima (5) kategori menjadi *class_index* dalam proses klasifikasi. Dataset terdiri dari 1000 *sample*, dan masing-masing *Class_Index* terdapat 200 *sample/instances*. Dataset yang digunakan dan komposisi masing-masing *Class_Index* disajikan pada Gambar 2.



Gambar 2. Komposisi Dataset berdasarkan Class_Index

2.2. *Preprocessing*

Pre-processing berguna untuk mengubah data teks yang tidak terstruktur menjadi data yang terstruktur [21] Proses *preprocessing* yang akan dilakukan pada penelitian meliputi lima tahapan yaitu *case folding*, *cleaning*, *tokenizing*, *filtering* dan *stemming*.

a. *Case folding* merupakan tahapan awal pada *preprocessing*, bertujuan untuk mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar (huruf kecil atau *lowercase*) [19]. Hasil dari proses *case folding* disajikan pada Tabel 1.

Tabel 1. Hasil *Case Folding*

Sebelum *Case Folding*	**Hasil *Case Folding***
Kasus positif virus Corona pada 24 April sebanyak 8.211 orang dan Jumlah pasien sembuh Corona di RI ada 1.002 orang dan meninggal 689 orang	kasus positif virus corona pada 24 april sebanyak 8.211 orang. jumlah pasien sembuh corona di ri ada 1.002 orang dan meninggal 689 orang

b. *Cleaning* merupakan proses pembersihan kata dengan menghilangkan angka, pembatas kata seperti koma (,), titik (.), dan tanda baca lainnya. Pembersihan kata bertujuan untuk mengurangi *noise* [21]. Hasil proses *cleaning* disajikan pada Tabel 2.

Tabel 2. Hasil *Cleaning*

Sebelum Cleaning	**Hasil Cleaning**
kasus positif virus corona pada 24 april sebanyak 8.211 orang. jumlah pasien sembuh corona di ri ada 1.002 orang dan meninggal 689 orang	kasus positif virus corona pada april sebanyak orang. jumlah pasien sembuh corona di ri ada orang dan meninggal orang

c. *Tokenizing* adalah pemotongan *string input* tiap kata penyusunnya. Pada tahap ini, karakter-karakter tertentu seperti tanda baca dihilangkan dan karakter spasi digunakan sebagai delimiter untuk memotong kalimat menjadi kumpulan kata. Hasil *Tokenizing* disajikan pada Tabel 3.

Tabel 3. Hasil *Tokenizing*

Sebelum *Tokenizing*	Hasil *Tokenizing*
kasus positif virus corona pada april sebanyak orang. jumlah pasien sembuh corona di ri ada orang dan meninggal orang	['sebelumnya', 'kasus', 'positif', 'virus', 'corona', 'pada', 'april', 'sebanyak', 'orang', 'jumlah', 'pasien', 'sembuh', 'corona', 'di', 'ri', 'ada', 'orang', 'dan', 'meninggal', 'orang']

d. *Filtering* adalah proses membuang kata yang tidak memilki makna atau tidak penting. Hasil proses *Filtering* disajikan pada Tabel 4.

Tabel 4. Pembagian data untuk *Training* dan *Testing*

Sebelum Filtering	**Hasil Filtering**
['sebelumnya', 'kasus', 'positif', 'virus', 'corona', 'pada', 'april', 'sebanyak', 'orang', 'jumlah', 'pasien', 'sembuh', 'corona', 'di', 'ri', 'ada', 'orang', 'dan', 'meninggal', 'orang']	['orang', 'corona', 'positif', 'virus', 'april', 'pasien', 'sembuh', 'ri', 'meninggal']

e. *Stemming* merupakan proses untuk mencari *stem* (kata dasar) dari kata hasil *stopword removal* (*filtering*). Hasil proses *Stemming* disajikan pada Tabel 5.

Tabel 5. Hasil *Stemming*

Sebelum *Stemming*	**Hasil *Stemming***
['orang', 'corona', 'positif', 'virus', 'april', 'pasien', 'sembuh', 'ri', 'meninggal']	['orang : orang', 'corona : corona', 'positif : positif', 'virus : virus', 'april : april', 'pasien : pasien', 'sembuh : sembuh', 'ri : ri', 'meninggal : tinggal']

2.3. *Word Embedding*

Word embedding atau *feature extraction* digunakan untuk memetakan kata ke vektor numerik yang dapat digunakan dalam komputasi. *One-hot encoding* adalah cara paling sederhana untuk mengkarakterisasi teks, dengan menggunakan panjang vektor untuk merepresentasikan sebuah kata. Posisi kata tersebut dipelajari dari teks atau berdasarkan kata-kata disekitarnya. *Word embedding* dapat menangkap makna semantik dan sintaktik kata. Model *word embedding* yang digunakan dalam penelitian ini adalah BoW, TF-IDF, Doc2Vec, dan Word2Vec. Ekstraksi fitur TF-IDF menggunakan bantuan *library Python3* yaitu TfidfVectorizer. Hasil dari TF-IDF diperoleh dari 1000 jumlah *reviews* memiliki 12387 kata. BoW menggunakan bantuan *library Python3 BoW_Vector*. Matrik yang dihasilkan dari proses BoW, 1000 *rows* x 16326 *columns*. Ekstraksi fitur Word2Vec, setiap kata yang ada pada *sample/instances* diproyeksikan ke suatu ruang dan membuat kata-kata dengan makna yang mirip akan berdekatan satu sama lain dalam ruang tersebut.

2.4. Klasifikasi Teks

Klasifikasi teks merupakan proses pengklasifikasian teks dengan algoritma *machine learning* (i.e. *Naïve Bayes*, *Support Vector Machines (SVM)*, *Decision Tree*, *K-Nearest Neighbors (KNN)*, *Random Forest*, *Logistic Regression*) dengan fitur *word embedding* (i.e. BoW, TF-IDF, Doc2Vec dan Word2Vec). Tahap ini menjelaskan mengenai proses pelatihan, validasi, dan pengujian pada klasifikasi teks. Penggunaan *word embedding* dijadikan fitur masukan dalam klasifikasi teks dengan melakukan percobaan dengan beberapa split data yang berbeda. *Split* data 50:50 (50% untuk data *training* dan 50% untuk data *testing*), *split* data 70:30 (70% untuk data *training* dan 30% untuk data *testing*), *split* data 80:20 (80% untuk data *training* dan 20% untuk data *testing*) dan *split* data 90:10 (90% untuk data *training* dan 10% untuk data *testing*).

2.5. *Confusion Matrix*

Confusion matrix multilabel terdiri dari lima kelas (kategori), menghitung nilai akurasi (*accuracy*) menggunakan persamaan 1, menghitung presisi (*precision*) menggunakan persamaan 2, menghitung *recall* menggunakan persamaan 3 dan *f1-score* menggunakan persamaan 4 dengan menghitung *True Positive (TPi)*, *False Negative (FNi)*, *False Positive (FPi)*, dan *True Negative (TNi)*, masing-masing kelas dan dibagi dengan jumlah kelas (1).

$$Akurasi = \frac{\Sigma_{i=1}^{l} \frac{TP_i + TN_i}{TP_i + FN_i + TN_i + FP_i}}{l} * 100\% \quad (1)$$

$$Presisi = \frac{Sigma_{i=1}^{l} TP_i}{\Sigma_{i=1}^{l}(FP_i + TP_i)} * 100\% \quad (2)$$

$$Recall = \frac{Sigma_{i=1}^{l} TP_i}{\Sigma_{i=1}^{l}(TP_i + FN_i)} * 100\% \quad (3)$$

$$F1score = \frac{2 * Presisi * Recall}{Presisi + Recall} \quad (4)$$

2.6. Analisa Hasil

Analisa hasil menjelaskan hasil dan analisa pengujian, dalam pengklasifikasian teks dalam mendapatkan model yang optimal. Selain itu, dalam tahap ini dilakukan proses analisis sistem interpretasi dengan mengukur *accuracy*, *precision*, *recall* dan *F1-score*.

3. HASIL DAN ANALISIS

Komparasi terhadap empat *feature extraction* (i.e. BoW, TF-IDF, Doc2Vec dan Word2Vec) dengan enam algoritma *machine learning* (i.e. NB, SVM, KNN, DT, RF dan LR), akan disajikan pada bagian ini. Kemudian dikategorikan dalam beberapa kategori yang ditunjukkan pada Tabel 6.

Tabel 6. Kategori Model Berdasarkan Nilai Kurva ROC [28]

Nilai *Accuracy*	Kategori
0.90 1.00	*Excellent Classification*
0.80 0.90	*Good Classification*
0.70 0.80	*Fair Classification*
0.60 0.70	*Poor Classification*
0.50 0.60	*Failure Classification*

3.1. Dataset

Dataset yang digunakan, diambil dari tribunnews.com pada tahun 2020-2021. Dataset merupakan sekumpulan dokumen berita dalam bentuk teks terdiri dari 1000 *sample/instances/records* disimpan dalam file tribunnewsE.csv. *Sample* berisi kata-kata dalam bahasa Indonesia dengan kisaran jumlah kata dalam satu *sample/instances* antara 300-500 kata per *sample/instances* nya. File tribunneswsE.csv ditampilkan, dalam bentuk *dataframe* melalui *library python import pandas as np*, hasilnya disajikan pada Gambar 3.

	Class Index	Category	Berita
0	4	Economy	(ADB) telah menyetujui untuk memberikan pinjaman berbasis kebijakan senilai US$ 500 juta atau setara dengan Rp 7,11 triliun (kurs Rp14.220 per dolar AS). Pinjaman tersebut untuk membantu Indonesia dalam meningkatkan kualitas sumber daya manusia, menaikkan produktivitas tenaga kerja, serta melakukan reformasi di bidang pendidikan, pengembangan keterampilan, kesehatan, dan perlindungan sosial. "Program baru ini akan membantu meningkatkan pembangunan sumber daya manusia, yang merupakan inti dari strategi pemerintah Indonesia dalam mencapai pertumbuhan ekonomi lebih tinggi dalam jangka panjang," kata Direktur ADB bidang Pembangunan Manusia dan Sosial bagi Asia Tenggara Ayako Inagaki, dikutip pada Senin (22/11). Selain itu, menurut Ayoko, program pinjaman ini juga untuk mendukung reformasi penting yang dapat membantu pemerintah untuk mencapai berbagai target kesehatan dan pendidikan dalam Tujuan Pembangunan Berkelanjutan Sustainable Development Goal (SDG). Hal tersebut seiring diperlukan tingkat pertumbuhan tahunan setidaknya 7% agar Indonesia mampu merealisasikan aspirasi menjadi negara berpenghasilan tinggi pada 2045, sehingga angkatan kerja yang terampil sangat penting bagi transisi Indonesia menuju manufaktur teknologi tinggi dan ekspor bernilai tambah lebih tinggi. Adapun, pemerintah Indonesia telah mengambil langkah-langkah untuk meningkatkan pembangunan sumber daya manusia. Tercatat indeks modal manusia Indonesia naik menjadi 54% pada 2020 dari sebelumnya 50% pada 2010. Pandemi Covid-19 pun berdampak negatif terhadap hasil pembelajaran. Hal ini akibat penutupan sekolah yang berkepanjangan, sehingga dalam jangka panjang berpengaruh bagi anak-anak yang masih kecil. Pandemi juga menyebabkan buruknya tingkat imunisasi bagi balita, karena perawatan kesehatan non Covid-19 menjadi lebih sulit diakses. Sejalan dengan dampak pandemi yang menekan permintaan dan memperlambat penciptaan lapangan kerja, pengangguran jangka panjang dapat menimbulkan terkikisnya keterampilan, terutama di kalangan kaum muda. "Dengan mengatasi defisit sumber daya manusia, program ini akan membantu meningkatkan pemulihan Indonesia dari pandemi global," kata Direktur ADB bidang Manajemen Publik, Sektor Keuangan, dan Perdagangan untuk Asia Tenggara Jose Antonio Tan III.
1	5	Otomotif	(ADM) berhasil membukukan 504 selama 11 hari penyelenggaraan pameran otomotif Gaikindo Indonesia International Auto Show (GIIAS) pada 11- 21 November 2021 di Indonesia Convention Exhibition (ICE)-BSD City, Tangerang. Hendrayadi Lastiyoso, Marketing & Customer Relations Division Head PT Astra International Daihatsu Sales Operation (AI-DSO) mengatakan angka perolehan tersebut turun 25 persen dibandingkan dengan gelaran GIIAS 2019 lalu. "Dalam ajang Pameran GIIAS 2021 ini pemesanan mobil Daihatsu yang diterima selama event 11 hari tersebut adalah sebanyak 504 unit. Jika dibandingkan dengan GIIAS 2019 yang lalu mengalami penurunan sekitar 25%. Penurunan ini dianggap sangat relevan seiring adanya pembatasan pengunjung dalam GIIAS kali ini," jelas Hendrayadi kepada Kontan, Senin (22/11/2021). Daihatsu juga berhasil melampaui target penjualan 500 unit dalam pameran ini. Penyumbang penjualan terbesar Daihatsu selama GIIAS 2021 diraih oleh Rocky dengan angka 108 unit atau berkontribusi sebanyak 27 persen. Penjualan terbesar lainnya ditempati oleh All New Xenia dengan total penjualan 104 unit dengan kontribusi 21%. Lalu Daihatsu Terios di posisi ketiga dengan kontribusi penjualan sebanyak 15 persen. Hendrayadi menyampaikan pula sebenarnya pihaknya tidak pernah benar-benar menetapkan target penjualan di GIIAS 2021. Daihatsu sendiri fokus memperkenalkan perkembangan teknologi yang dikembangkan oleh Daihatsu kepada publik. "Tujuan Daihatsu di GIIAS tahun ini adalah memperkenalkan model baru dari Xenia, yaitu All New Xenia. Maka meski di booth Daihatsu menerima pemesanan sebanyak 504 unit maka hal itu kami anggap sebagai bonus tambahan dari tujuan yang ingin dicapai," paparnya. Dengan pencapaian di pameran GIIAS 2021 ini, Daihatsu optimistis bisa mempertahankan market share nomor 2 minimal sebesar 17% hingga akhir tahun. Ia menambahkan, hingga Oktober lalu, saja Daihatsu bisa meraih market share 17,3% sehingga pihaknya bisa mempertahankannya hingga akhir ini. Daihatsu memandang tren kenaikan positif di industri otomotif tahun 2022 makin marak. "Jika dilihat periode Januari sampai dengan Oktober tahun ini, market otomotif Indonesia terus mengalami trend kenaikkan yang positif. Maka Daihatsu

Gambar 3. Dataset Tribunnews.com

3.2. *Preprocessing*

Sebagai langkah awal dari proses klasifikasi teks multilabel adalah tahap *text preprocessing*, dimana tiap dataset akan disiapkan terlebih dahulu sebelum menerapkan *feature extraction*. Tahapan yang digunakan pada *text preprocessing*, yaitu: *case folding*, *stopwords removal*, *stemming*, dan *tokenization*. Secara garis besar, langkah yang dilakukan untuk tiap dokumennya adalah:

a. Membaca isi file dan menyimpannya ke dalam *variable*
b. Melakukan *preprocessing* terhadap *variable* tersebut
c. Menyimpan isi *variable* ke dalam file kembali

Hasil dari proses *preprocessing* ini adalah sekumpulan dataset yang telah bersih. Hasil dari *preprocessing* ini adalah sekumpulan dataset yang telah bersih. Penggabungan hasil *text preprocessing* disajikan pada Gambar 4.

	Berita	SubKategori	Kategori	tokenizing	stopword	stemming	berita_clean
0	pihak kepolisian terus melakukan pendalaman te...	Nasional	News	[pihak, kepolisian, terus, melakukan, pendalam...	[kepolisian, pendalaman, terkait, oknum, lurah...	[polisi, dalam, kait, oknum, lurah, kota, gela...	polisi dalam kait oknum lurah kota gelar hajat...
1	berikut ini penjelasan mengenai pendaftaran ka...	Nasional	News	[berikut, ini, penjelasan, mengenai, pendaftar...	[penjelasan, pendaftaran, kartu, prakerja, gel...	[jelas, daftar, kartu, prakerja, gelombang, le...	jelas daftar kartu prakerja gelombang lengkap ...
2	ketua dpr ri puan maharani meninjau pelaksanaa...	Nasional	News	[ketua, dpr, ri, puan, maharani, meninjau, pel...	[ketua, dpr, ri, puan, maharani, meninjau, pel...	[ketua, dpr, ri, puan, maharani, tinjau, laksa...	ketua dpr ri puan maharani tinjau laksana vaks...
3	pengadilan menghukum dari orang yang dituduh m...	Nasional	News	[pengadilan, menghukum, dari, orang, yang, dit...	[pengadilan, menghukum, orang, dituduh, remaja...	[adil, hukum, orang, tuduh, remaja, video, ant...	adil hukum orang tuduh remaja video antislam ...
4	berikut adalah daftar formasi calon pegawai ne...	nasional	News	[berikut, adalah, daftar, formasi, calon, pega...	[daftar, formasi, calon, pegawai, negeri, sipi...	[daftar, formasi, calon, pegawai, negeri, sipi...	daftar formasi calon pegawai negeri sipil cpns...

Gambar 4. Hasil penggabungan *text preprocessing*

3.3. Akurasi Model Klasifikasi Teks

Setelah *feature extraction* atau *word embedding*, dilakukan klasifikasi dengan menggunakan algoritma *machine learning*, seperti: *Naïve Bayes (NB)*, *Support Vector Machine (SVM), Decision Tree (DT), K-Nearest Neighbors (KNN), Random Forest (RF),* dan *Logistic Regression (LR)*. Selanjutnya, akurasi dari masing-masing model dibandingkan satu dengan yang lainnya. Akurasi menggambarkan seberapa besar tingkat akurat model yang telah dibuat dapat mengklasifikasi data dengan benar. Akurasi didapatkan dari perhitungan rasio prediksi benar dengan keseluruhan data. Berikut merupakan hasil akurasi model dari pelatihan dan pengujian klasifikasi teks dengan ekstraksi fitur (*feature extraction*) yang berbeda

1. TF-IDF

Tabel 7 menyajikan nilai akurasi dari algoritma *machine learning* dengan ekstraksi fitur TF-IDF. Algoritma *Naïve Bayes*, akurasi tertinggi terdapat pada *split* data 90:10 (90% data *training* dan 10% data *testing*) sebesar 80% (0.87). Algoritma SVM, akurasi tertinggi terdapat pada *split* data 90:10, sebesar 86% (0.86). Algoritma *Decision Tree*, akurasi model tertinggi pada *split* data 80:20 dan 90:10, sebesar 69% (0.69). Algoritma KNN, model yang memiliki akurasi tertinggi terdapat pada *split* data 80:20 dan 90:10 sebesar 81% (0.81). Algoritma *Random Forest*, akurasi tertinggi ada pada *split* data 90:10 sebesar 81% (0.81). Sedangkan algoritma *Logistic Regression*, akurasi tertinggi terdapat pada split data 90:10 sebesar 85% (0.85). Dari keseluruhan model klasifikasi teks berdasarkan kategori berita dengan menggunakan ekstraksi fitur TF-IDF, algoritma *machine learning* yang memiliki akurasi tertinggi adalah *Naïve Bayes* pada *split* data 90:10 sebesar 87%.

Penerapan metode *Naïve Bayes Classifier* dalam klasifikasi berita (teks) memiliki akurasi yang baik terbukti pada data uji yang bersumber dari situs web (i.e. tribunnews.com) menghasilkan nilai akurasi dengan persentase yang tinggi yaitu 87% untuk data *training* yang besar (900 artikel). Akurasi semakin tinggi dengan meningkatnya data *training* yang digunakan dalam pembelajaran dengan perbandingan data *training* dan data *testing* dengan *split* data 90:10.

Tabel 7. Akurasi algoritma *machine learning* dengan TF-IDF

Split Data	**Algoritma *Machine Learning* + TF-IDF**					
	Naïve Bayes	**SVM**	***Decision Tree***	**KNN**	***Random Forest***	***Logistic Regression***
50:50	80.8	77.8	63.2	76.2	79.6	81.8
70:30	80.6	83	64	79	81.3	81.33
80:20	82.5	83	69	81	79.5	81.5
90:10	87	86	69	81	82	85

2. BoW

Tabel 8 menyajikan hasil akurasi model klasifikasi teks menggunakan algoritma *machine learning* (i.e. *Naïve Bayes*, SVM, *Decision Tree*, KNN, *Random Forest* dan *Logistic Regression*) dengan ekstraksi fitur BoW. Berdasarkan nilai yang ada pada Tabel 2, algoritma *Naïve Bayes* memiliki akurasi tertinggi pada *split* data 70:30 sebesar 83% (0.83). Algoritma SVM, akurasi tertinggi pada *split* data 70:30 sebesar 81.3% (0.813). Algoritma *Decision Tree*, akurasi tertinggi pada *split* data 80:20 sebesar 67% (0.67). Algoritma KNN, akurasi tertinggi pada *split* data 70:30 sebesar 57.67% (0.5767). Algoritma *Random Forest*, akurasi tertinggi pada *split* data 80:20 sebesar 80% (0.80). Sedangkan algoritma *Logistic Regression*, akurasi tertinggi pada *split* data 80:20 dan 90:10 sebesar 82%. Dari keseluruhan model

klasifikasi teks menggunakan ekstraksi fitur BoW dengan beberapa *split* data yang berbeda, algoritma yang memiliki akurasi tertinggi adalah *Naïve Bayes* pada *split* data 70:30 sebesar 83%.

Tabel 8. Akurasi algoritma *machine learning* dengan BoW

Split Data	**Algoritma *Machine Learning* + BoW**					
	Naïve Bayes	**SVM**	***Decision Tree***	**KNN**	***Random Forest***	***Logistic Regression***
50:50	78.4	77.8	63.80	47.59	76.0	79.0
70:30	83.0	81.3	66.33	57.67	77.67	81.33
80:20	80.5	81.0	67.0	57.49	80.0	82.0
90:10	78.0	75.0	62.0	43.0	79.0	82.0

3. Doc2Vec

Tabel 9 menyajikan hasil akurasi model klasifikasi teks menggunakan algoritma *machine learning* (i.e. *Naïve Bayes*, SVM, *Decision Tree*, KNN, *Random Forest* dan *Logistic Regression*) dengan ekstraksi fitur Doc2Vec. Algoritma *Naïve Bayes* memiliki akurasi tertinggi pada *split* data 90:10 sebesar 70%. Algoritma SVM memiliki akurasi tertinggi pada *split* data 90:10 sebesar 81% (0.81). Algoritma *Decision Tree*, akurasi tertinggi pada *split* data 90:10 sebesar 78%. Algoritma KNN, akurasi tertinggi pada *split* data 80:20 sebesar 75% (0.75). Algoritma *Random Forest*, akurasi tertinggi pada *split* data 90:10 sebesar 78% (0.78). Algoritma *Logistic Regression*, akurasi tertinggi pada *split* data 90:10 sebesar 76% (0.76). Dari keseluruhan model klasifikasi teks pada kategori berita menggunakan ekstraksi fitur Doc2Vec, algoritma *machine learning* yang memiliki akurasi tertinggi adalah SVM pada *split* data 90:10 sebesar 81%.

Tabel 9. Akurasi algoritma *machine learning* dengan Doc2Vec

Split Data	**Algoritma *Machine Learning* + Doc2Vec**					
	Naïve Bayes	**SVM**	***Decision Tree***	**KNN**	***Random Forest***	***Logistic Regression***
50:50	35.4	35.8	51.6	50.6	60.19	46.2
70:30	53.7	70.0	65.0	68.3	71.0	71.0
80:20	62.0	74.5	67.5	75.0	73.5	70.5
90:10	70.0	81.0	78.0	71.0	78.0	76.0

4. Word2Vec

Tabel 4 merupakan hasil akurasi model klasifikasi teks menggunakan algoritma *Machine Learning* (i.e. *Naïve Bayes*, SVM, *Decision Tree*, KNN, *Random Forest* dan *Logistic Regression*) dengan ekstraksi fitur Word2Vec. Dari data yang disajikan pada Tabel 10, akurasi model klasifikasi teks multilabel (kategori berita) dari ke enam algoritma *machine learning* menggunakan ekstraksi fitur Word2Vec tidak bekerja secara optimal. Semua akurasi model berada dibawah 50%.

Ini menunjukkan bahwa ekstraksi fitur Word2Vec kurang optimal apabila digunakan dalam algoritma *machine learning* khususnya melakukan klasifikasi teks multilabel (kategori berita) pada dataset tribunnews dengan banyaknya 1000 sample. Word2Vec dikembangkan merupakan salah satu aplikasi *unsupervised learning* menggunakan *neural network* yang terdiri dari sebuah *hidden layer* dan *fully connected layer*. Rendahnya akurasi model, juga bisa disebabkan oleh jumlah dataset yang digunakan. Karena dalam jumlah dataset yang sedikit Word2vec tidak dapat menangkap kemiripan makna kata dengan baik.

Tabel 10. Akurasi algoritma *machine learning* dengan Word2Vec

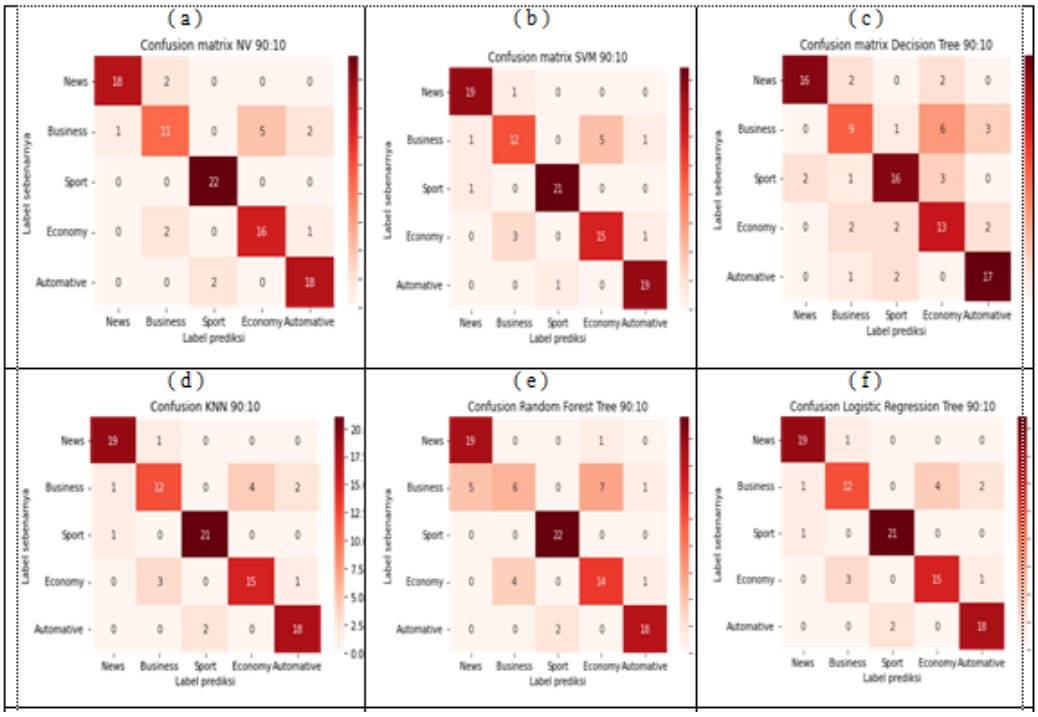
Split Data	**Algoritma Machine Learning + Word2Vec**					
	Naïve Bayes	**SVM**	***Decision Tree***	**KNN**	***Random Forest***	***Logistic Regression***
50:50	23	26	23	21	29	19
70:30	23	23	26	22	29	16
80:20	23	18	22	20	33	20
90:10	21	22	24	14	32	24

3.4. Evaluasi dan Validasi Model Klasifikasi Teks

Setelah dilakukan percobaan klasifikasi teks menggunakan algoritma *machine learning* dan ekstraksi fitur, maka dilakukan pengujian tingkat akurasi pada tiap-tiap model. Evaluasi tingkat akurasi algoritma dilakukan dengan menggunakan *confusion matrix* dan kurva AUC (*Area Under Curva*). AUC memiliki tingkat nilai *diagnose*, model dikatakan (i) *Excellent classification* jika memiliki akurasi 0.90-1.00. (ii) *Good classification*, jika memiliki akurasi 0.80-0.90, (iii) *Fair Classification*, jika memiliki akurasi 0.70-0.80, (iv) Poor *classification*, jika memiliki akurasi 0.60-0.70, dan (v) Failure Classification, jika memiliki akurasi 0.50-0.60. *Confusion matrix* memberikan informasi perbandingan hasil klasifikasi yang dilakukan oleh model yang telah dilatih dengan hasil klasifikasi sebenarnya. Karena keterbatasan *space*, *confusion matrix* dan hasil evaluasi kinerja proses pengujian yang ditampilkan pada bagian ini adalah algoritma *machine learning* yang memiliki akurasi tertinggi pada masing-masing ekstraksi fitur dari keseluruhan *split* data.

1. TF-IDF

Dari percobaan yang dilakukan pada ke enam algoritma *Machine Learning* dengan fitur ekstraksi TF-IDF, diperoleh model akurasi pelatihan dengan *good classification* adalah NB sebesar 0.87 pada *split* data 90:10; SVM sebesar 0.86 pada *split* data 90:10, KNN sebesar 0.85 pada *split* data 90:10, RF sebesar 0.82 pada *split* data 90:10 dan LR sebesar 0.85 pada *split* data 90:10. Sedangkan algoritma DT dikategorikan sebagai model *poor classification*, karena memilki akurasi tertinggi sebesar 0.69 pada *split* data dan 90:10. *Confusion matrix* dari ke enam algoritma *machine learning* (akurasi tertinggi) dan ekstraksi fitur TF-IDF dengan *split* data 90:10, disajikan pada Gambar 5. Gambar 5 (a) visualisasi confusion matrix dari NB, Gambar 5 (b) merupakan confusion matrix dari SVM, Gambar 5 (c) merupakan *confusion matrix Decision Tree*, Gambar 5 (d) merupakan *confusion matrix* KNN, Gambar 5 (e) visuaisasi *confusion matrix Random Forest*, dan Gambar 5 (f) merupakan *confusion matrix Logistic Regression*.



Gambar 5. *Confusion Matrix* Algoritma *Machine Learning* dan Ekstraksi Fitur ”TF-IDF”

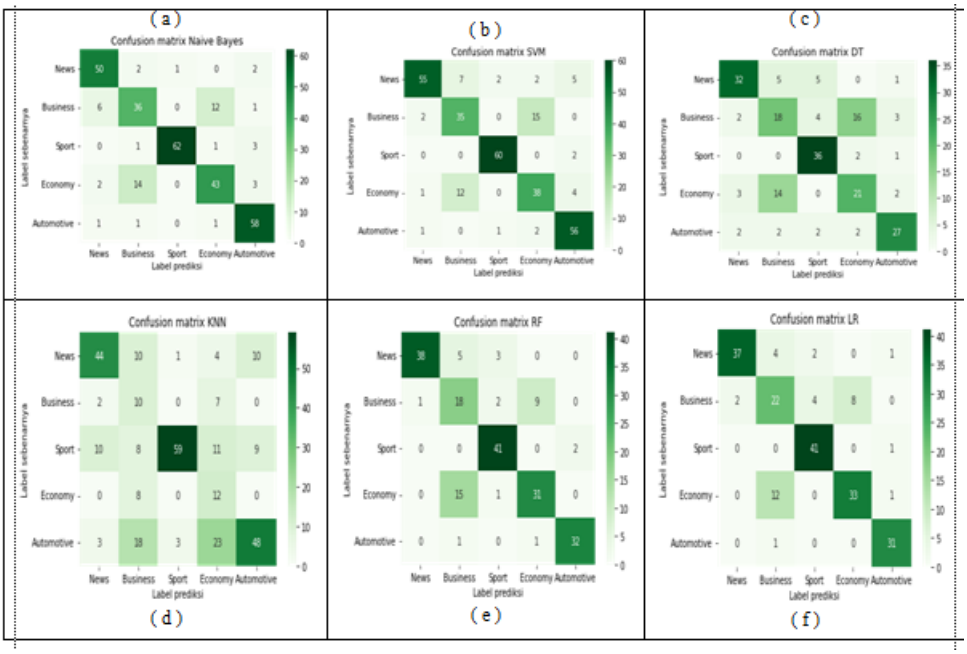
Tabel 11. Hasil Kinerja Evaluasi Pengujian Model dan Ekstraksi Fitur ”TF-IDF”

Model	Label	Precision	*Recall*	*f1-score*	*Support*	Model	Label	*Precision*	*Recall*	*f1-score*	Support
	News	0.94	0.85	0.89	8		*News*	0.89	0.80	0.84	20
NB	*Business*	0.73	0.58	0.65	22	**KNN**	*Business*	0.60	0.47	0.53	19
	Sport	0.96	1.00	0.98	23		*Sport*	0.76	0.73	0.74	22
***Split* data 90:10**	*Economy*	0.74	0.89	0.81	27	***Split* data 90:10**	*Economy*	0.54	0.68	0.60	19
	Automotive	0.90	0.95	0.93	20		Automotive	0.77	0.85	0.81	20
Akurasi Pengujian				0.85	100	**Akurasi Pengujian**				0.85	100
	News	0.90	0.95	0.93	20		*News*	0.79	0.95	0.86	20
SVM	*Business*	0.75	0.63	0.69	19	**RF**	*Business*	0.60	0.32	0.41	19
	Sport	0.95	0.95	0.95	22		*Sport*	0.92	1.00	0.96	22
***Split* data 90:10**	*Economy*	0.75	0.79	0.77	19	***Split* data 90:10**	*Economy*	0.64	0.74	0.68	19
	Automotive	0.90	0.95	0.93	20		*Automotive*	0.90	0.90	0.90	20
Akurasi Pengujian				0.84	100	**Akurasi Pengujian**				0.80	100
	News	0.89	0.80	0.84	20		*News*	0.90	0.95	0.93	20
DT	*Business*	0.60	0.97	0.53	19	**LR**	*Business*	0.79	0.58	0.67	19
	Sport	0.76	0.73	0.74	22		*Sport*	0.91	0.95	0.93	22
Split data 90:10	*Economy*	0.54	0.68	0.60	19	**Split data 90:10**	*Economy*	0.76	0.84	0.80	19
	Automotive	0.77	0.65	0.81	20		*Automotive*	0.86	0.90	0.88	20
Akurasi Pengujian				0.70	100	**Akurasi Pengujian**				0.85	100

Dari Tabel 11, evaluasi pengujian yang dilakukan pada tiap-tiap model dengan *split* data 90:10, dimana 90% merupakan data *training* (900 *sample*) dan 10% data *testing* (100 *sample*), nilai akurasi pengujian yang diperoleh adalah NB sebesar 0.85, SVM sebesar 0.84, DT sebesar 0.70, KNN sebesar 0.85, RF sebesar 0.80 dan LR sebesar 0.85. Hal ini menunjukkan bahwa akurasi model pada saat pelatihan dan pengujian tidak jauh berbeda, bahkan hampir sama. Berdasarkan AUC, lima algoritma (i.e. NB, SVM, KNN, RF, dan LR) merupakan kategori *good classification* untuk melakukan klasifikasi berita khususnya pada tribunnews.com, sedangkan DT merupakan *poor classification*.

2. BoW

Nilai akurasi tertinggi pada saat pelatihan dari ke enam algoritma *machine learning* dan ekstraksi fitur BoW adalah NB pada *split* data 70:30 sebesar 0.83, SVM pada *split* data 70:30 sebesar 0.813, DT pada *split* data 80:20 sebesar 0.67, KNN pada *split* data 70:30 sebesar 0.677, RF pada *split* data 80:20 sebesar 0.80, dan LR pada *split* data 80:20 dan 90:10 sebesar 0.81. Empat model (NB, SVM, RF dan LR) termasuk pada *good classification* berdasarkan AUC, sedangkan dua algoritma (KNN dan DT) merupakan *poor classification*. Hasil *confusion matrix* klasifikasi teks dari ke enam algoritma *machine learning* dan ekstraksi fitur BoW dengan akurasi tertinggi, disajikan pada Gambar 6. Gambar 6(a) visualisasi *confusion matrix* dari NB, Gambar 6(b) merupakan *confusion matrix* dari SVM, Gambar 6(c) merupakan *confusion matrix Decision Tree*, Gambar 6(d) merupakan *confusion matrix* KNN, Gambar 6(e) visuaisasi *confusion matrix Random Forest*, dan Gambar 6(f) merupakan *confusion matrix Logistic Regression*.



Gambar 6. *Confusion Matrix* dari Algoritma *Machine Learning* dan Ekstraksi Fitur "BoW"

Hasil *confusion matrix* yang disajikan pada Gambar 5, dari 1000 *sample* yang diberikan kepada enam algoritma *machine learning* (i.e. NB, SVM, DT, KNN, RF, dan LR) dengan *split* data yang berbeda. Akurasi tertinggi dari model pada saat pelatihan adalah NB dengan 70% data *training* (700 *sample*) dan 30% data *testing* (300 *sample*). Hasil *confusion matrix*-nya, seperti yang disajikan pada Gambar 5(a), menunjukkan bahwa aktualnya merupakan kategori *News* dan hasil prediksinya juga *News* sebanyak 50 *sample*; aktualnya kategori *Business* dan hasil prediksinya juga *Business* sebanyak 36 *sample*; aktualnya *Sport* dan hasil prediksi juga *Sport* sebanyak 62 *sample*; aktualnya *Economy* dan hasil prediksinya juga *Economy* sebanyak 43 *sample*; nilai aktualnya *Automative* dan prediksinya juga *Automotive* adalah sebanyak 58 *sample*. Evalusi kinerja model dari ke enam algoritma *machine learning* yang memilki nilai akurasi pelatihan tertinggi dengan ekstraksi fitur BoW disajikan pada Tabel 6. Nilai *accuracy*, *precision*, *recall* dan *f1-score* yang disajikan pada Tabel 12, berasal dari nil TP, TN, FP dan FN dari *confusion matrix* yang ada pada Gambar 6.

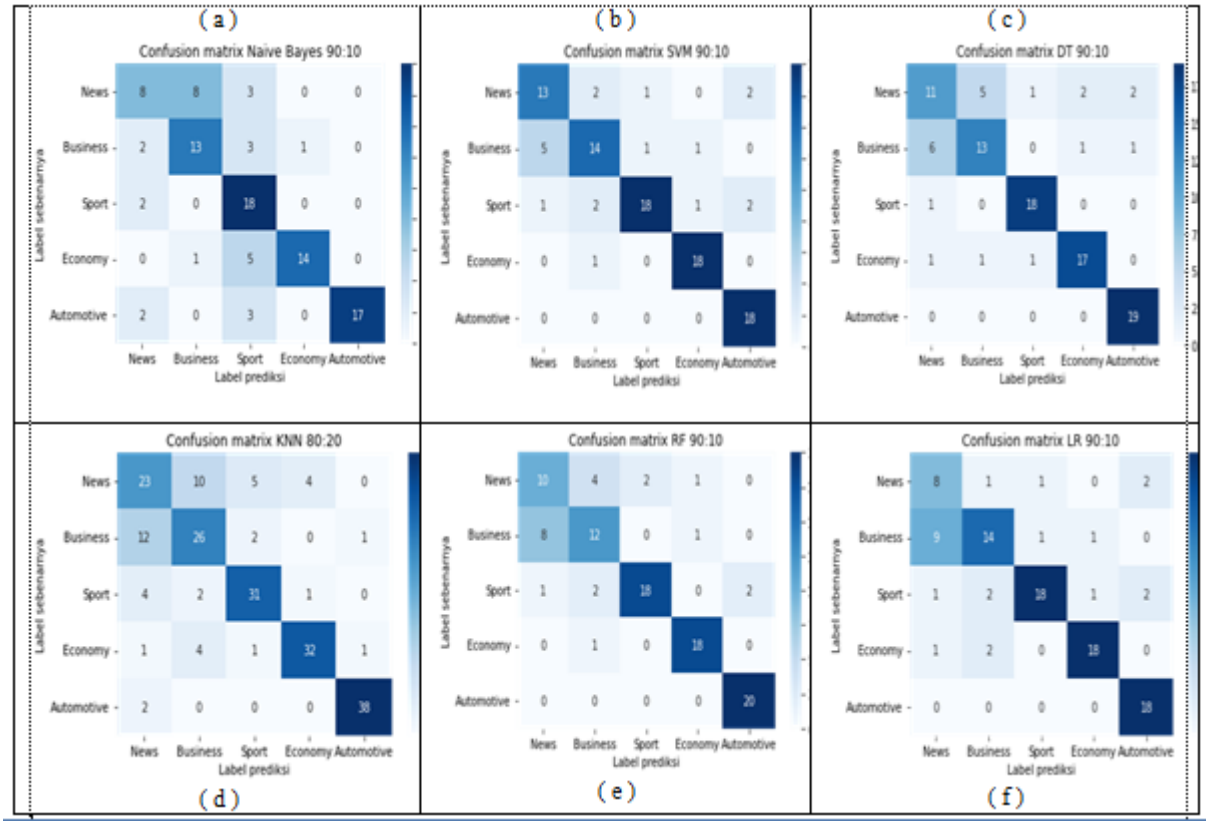
Tabel 12. Hasil Kinerja Evaluasi Pengujian Model dan Ekstraksi Fitur "BoW"

Model	Label	*Precision*	*Recall*	*f1-score*	*Support*	Model	Label	*Precision*	*Recall*	*f1-score*	*Support*
	News	0.85	0.91	0.88	55		*News*	0.75	0.64	0.73	69
NB	*Business*	0.67	0.66	0.66	55	**KNN**	*Business*	0.19	0.41	0.40	19
	Sport	0.98	0.93	0.93	67		*Sport*	0.94	0.92	0.8	97
***Split* data 70:30**	*Economy*	0.75	0.69	0.69	62	***Split* data 70:30**	*Economy*	0.21	0.53	0.54	20
	Automotive	0.87	0.95	0.95	61		*Automotive*	0.72	0.79	0.77	95
Akurasi Pengujian				0.83	300	**Akurasi Pengujian**				0.58	300
	News	0.93	0.77	0.85	71		News	0.97	0.83	0.89	46
SVM	*Business*	0.65	0.67	0.66	52	**RF**	*Business*	0.46	0.60	0.52	30
	Sport	0.95	0.97	0.96	62		Sport	0.87	0.95	0.91	43
Split data 70:30	*Economy*	0.67	0.69	0.68	55	***Split* data 80:20**	Economy	0.76	0.66	0.70	47
	Automotive	0.84	0.93	0.88	60		Automotive	0.94	0.94	0.94	34
Akurasi Pengujian				0.81	300	**Akurasi Pengujian**				0.80	200
	News	0.82	0.74	0.78	43		*News*	0.95	0.84	0.89	44
DT	*Business*	0.46	0.42	0.44	43	**LR**	*Business*	0.56	0.61	0.59	36
	Sport	0.77	0.92	0.84	39		*Sport*	0.87	0.98	0.92	42
Split data 80:20	*Economy*	0.51	0.53	0.52	40	***Split* data 80:20**	Economy	0.80	0.72	0.76	46
	Automotive	0.70	0.77	0.78	35		*Automotive*	0.91	0.97	0.94	32
Akurasi Pengujian				0.67	200	**Akurasi Pengujian**				0.82	200

Tabel 6 merupakan hasil pengujian dari enam algoritma *machine learning* (i.e. NB, SVM, KNN, DT, RF, dan LR) dengan ekstraksi fitur BoW. Dari pengujian diperoleh nilai *accuracy*, *precision*, *recall* dan *f1-score* dari tiap-tiap model. Akurasi pengujian dan akurasi pelatihan tidak berbeda. Berdasarkan hasil diagnose dari AUC, algoritma NB, SVM, RF dan LR memilki nilai akurasi besar sama dengan (¿=) 0.80 yang merupakan kategori *good classification* untuk melakukan klasifikasi teks berdasarkan kategori berita dalam dataset tribunnews.com. Algoritma DT memilki nilai akurasi pengujian sebesar 0.67 dengan 80% data *training* atau 800 *sample* dan 20% data *testing* dari 1000 *sample*, termasuk pada *poor classification* untuk melakukan klasifikasi teks berdasarkan kategori berita pada dataset tribunnews.com. Sedangkan algoritma KNN dengan akurasi pengujian sebesar 0.58, berdasarkan AUC termasuk kategori *failure classification*. Dari empat algoritma *machine learning* (i.e. NB, SVM, RF dan LR), yang memiliki akurasi pengujian tertinggi adalah NB, artinya model NB+BoW lebih baik dari model yang lain untuk melakukan klasifikasi teks sesuai dengan kategori berita yang ada pada dataset tribunnews.com.

3. Doc2Vec

Nilai akurasi tertinggi dari percobaan yang dilakukan pada ke enam algoritma *machine learning* dan ekstraksi fitur Doc2Vecadalah NB pada *split* data 90:10 sebesar 0.70, SVM pada *split* data 90:10 sebesar 0.81, DT pada *split* data 90:10 sebesar 0.78, KNN pada *split* data 80:20 sebesar 0.75, RF pada *split* data 90:10 sebesar 0.78, dan LR pada *split* data 90:10 sebesar 0.76. Berdasarkan hasil analisa dengan AUC, hanya satu algoritma *machine learning* (i.e. SVM) dengan ekstraksi fitur Doc2Vec yang masuk ke dalam kategori *good classification*. Sedangkan lima algoritma *machine learning* (i.e. NB, DT, KNN, RF, dan LR) masuk ke dalam kategori disajikan *poor classification*. *Confusion matrix* klasifikasi teks dari ke enam algoritma *machine learning* dan ekstraksi fitur BoW dengan akurasi tertinggi, disajikan pada Gambar 7. Gambar 7(a) visualisasi *confusion matrix* dari NB, Gambar 7(b) merupakan *confusion matrix* dari SVM, Gambar 7(c) merupakan *confusion matrix Decision Tree*, Gambar 7(d) merupakan *confusion matrix* KNN, Gambar 7(e) visualisasi *confusion matrix Random Forest*, dan Gambar 7(f) merupakan *confusion matrix Logistic Regression*.



Gambar 7. *Confusion Matrix* dari Algoritma *Machine Learning* dan Ekstraksi Fitur "Doc2Vec"

Hasil *confusion matrix* yang disajikan pada Gambar 6, terhadap 1000 *sample* yang diberikan kepada enam algoritma *machine learning* (i.e. NB, SVM, DT, KNN, RF, dan LR) dengan beberapa *split* data yang berbeda. Akurasi tertinggi dari model pada saat pelatihan adalah algoritma SVM dengan 90% data *training* (900 *sample*) dan 10% data *testing* (100 *sample*). Hasil *confusion matrix*-nya, seperti yang disajikan pada Gambar 6(b), menunjukkan bahwa aktualnya merupakan kategori *News* dan hasil prediksinya juga *News* sebanyak 13 *sample*; aktualnya kategori *Business* dan hasil prediksinya juga *Business* sebanyak 14 *sample*; aktualnya *Sport* dan hasi prediksi juga *Sport* sebanyak 18 *sample*; aktualnya *Economy* dan hasil prediksinya juga *Economy* sebanyak 18 *sample*; nilai aktualnya *Automotive* dan prediksinya juga *Automotive* adalah sebanyak 18 *sample*. Evalusi kinerja model dari ke enam algoritma *machine learning* yang memiliki nilai akurasi pelatihan tertinggi dengan ekstraksi fitur Doc2Vec disajikan pada Tabel 13. Nilai *accuracy*, *precision*, *recall* dan *f1-score* yang disajikan pada Tabel 6, berasal dari nilai TP, TN, FP dan FN dari *confusion matrix* yang ada pada Gambar 7.

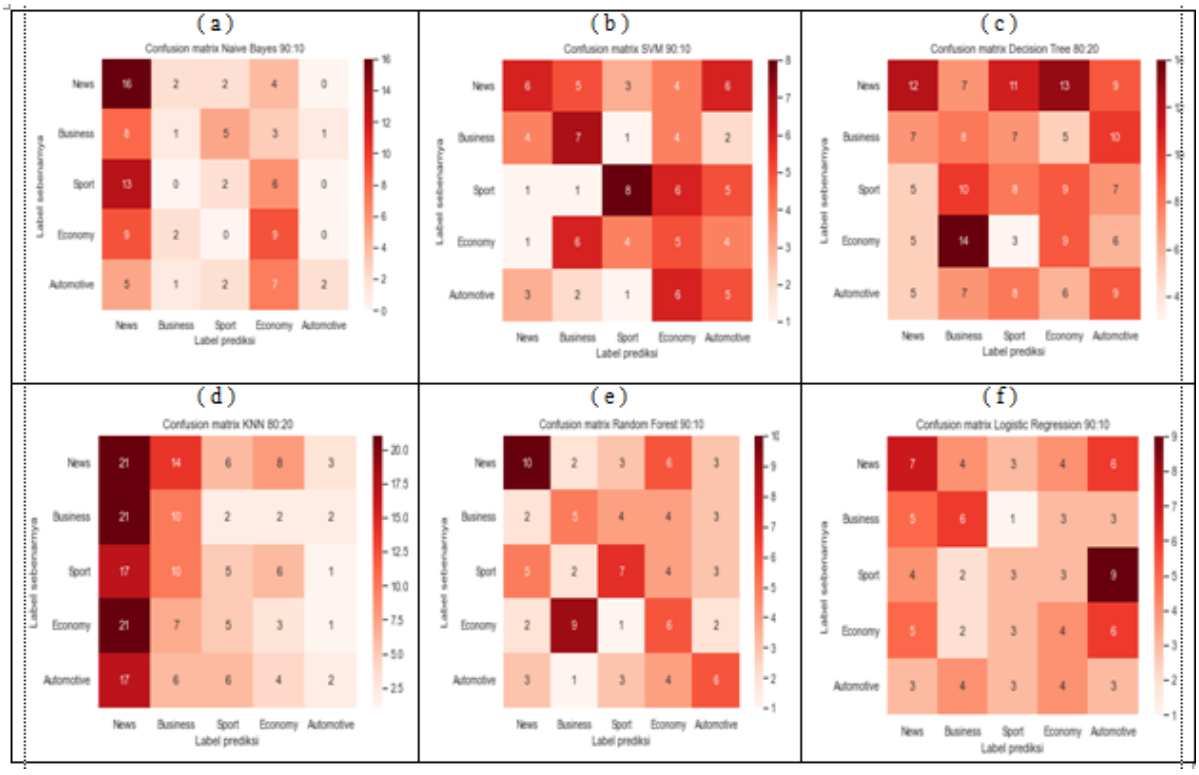
Tabel 13. Hasil Kinerja Evaluasi Pengujian Model dan Ekstraksi Fitur ”Doc2Vec”

Model	Label	*Precision*	*Recall*	*f1-score*	*Support*	Model	Label	*Precision*	*Recall*	*f1-score*	*Support*
NB ***Split* data 90:10**	*News*	0.57	0.42	0.48	19	**KNN** ***Split* data 80:20**	*News*	0.55	0.55	0.55	42
	Business	0.59	0.68	0.63	19		*Business*	0.62	0.63	0.63	42
	Sport	0.56	0.90	0.69	20		*Sport*	0.79	0.82	0.81	38
	Economy	0.93	0.70	0.80	20		*Economy*	0.86	0.82	0.84	39
	Automotive	1.00	0.77	0.87	22		*Automotive*	0.95	0.95	0.95	40
Akurasi Pengujian				0.70	100	**Akurasi Pengujian**				0.75	200
SVM Split data 90:10	*News*	0.68	0.72	0.70	18	**RF** ***Split* data 90:10**	*News*	0.53	0.59	0.56	17
	Business	0.74	0.67	0.70	21		*Business*	0.63	0.57	0.60	21
	Sport	0.90	0.75	0.82	24		*Sport*	0.90	0.78	0.84	23
	Economy	0.90	0.95	0.92	19		*Economy*	0.90	0.95	0.92	19
	Automotive	0.82	1.00	0.90	18		*Automotive*	0.91	1.00	0.95	20
Akurasi Pengujian				0.81	100	**Akurasi Pengujian**				0.78	100
DT ***Split* data 90:10**	*News*	0.58	0.52	0.55	21	**LR** ***Split* data 90:10**	*News*	0.42	0.67	0.52	12
	Business	0.68	0.62	0.65	21		*Business*	0.74	0.56	0.64	25
	Sport	0.90	095	0.92	19		*Sport*	0.90	0.75	0.82	24
	Economy	0.90	0.85	0.85	20		*Economy*	0.90	0.86	0.88	21
	Automotive	0.91	1.00	0.93	19		*Automotive*	0.82	1.00	0.90	18
Akurasi Pengujian				0.78	100	**Akurasi Pengujian**				0.76	100

Tabel 13 merupakan hasil pengujian dari enam algoritma *machine learning* (i.e. NB, SVM, KNN, DT, RF, dan LR) dengan ekstraksi fitur Doc2Vec. Dari pengujian diperoleh nilai *accuracy*, *precision*, *recall* dan *f1-score* dari tiap tiap model. Akurasi pengujian dan akurasi pelatihan sama. Berdasarkan AUC, algoritma SVM memiliki nilai akurasi pengujian 0.81 yang merupakan kategori *good classification* untuk melakukan klasifikasi teks berdasarkan kategori berita dalam dataset tribunnews.com. Sedangkan algoritma NB,DT, KNN, RF, dan LR memiliki nilai akurasi pengujian antara 0.60 0.70, termasuk pada *poor classification* untuk melakukan klasifikasi teks berdasarkan kategori berita pada dataset tribunnews.com. Dari enam algoritma *machine learning* (i.e. NB, SVM, DT, KNN, RF dan LR), yang memiliki akurasi pengujian tertinggi adalah SVM, artinya model SVM + Doc2Vec lebih baik dari model yang lain untuk melakukan klasifikasi teks sesuai dengan kategori berita yang ada pada dataset tribunnews.com.

4. Word2Vec

Nilai *accuracy* pelatihan tertinggi pada klasifikasi teks menggunakan algoritma *machine learning* (i.e. NB, SVM, DT, KNN, RF dan LR) dan ekstraksi fitur Word2Vecadalah NB pada *split* data 90:10 sebesar 0.3, SVM pada *split* data 90:10 sebesar 0.29, DT pada *split* data 80:20 sebesar 0.22, KNN pada *split* data 80:20 sebesar 0.23, RF pada *split* data 90:10 sebesar 0.33 dan LR pada *split* data 90:10 sebesar 0.23. Hasil dari AUC, ke enam algoritma *machine learning* dengan ekstraksi fitur Word2Vec merupakan kategori *failure classification*. *Confusion matrix* dari ke enam algoritma *machine learning* dengan ekstraksi fitur Word2Vec disajikan pada Gambar 8.



Gambar 8. *Confusion Matrix* dari Algoritma *Machine Learning* dan Ekstraksi Fitur ”Word2Vec”

Hasil *confusion matrix* yang disajikan pada Gambar 8, terhadap 1000 *sample* yang diberikan kepada enam algoritma *machine learning* (i.e. NB, SVM, DT, KNN, RF, dan LR) dengan beberapa *split* data yang berbeda. Akurasi tertinggi dari model pada saat pelatihan adalah algoritma RF dengan 90% data *training* (900 *sample*) dan 10% data *testing* (100 *sample*). Hasil *confusion matrixnya*, seperti yang disajikan pada Gambar 7(e), menunjukkan bahwa aktualnya merupakan kategori *News* dan hasil prediksinya juga *News* sebanyak 10 *sample*; aktualnya kategori *Business* dan hasil prediksinya juga *Business* sebanyak 5 *sample*; aktualnya *Sport* dan hasi prediksi juga *Sport* sebanyak 7 *sample*; aktualnya *Economy* dan hasil prediksinya juga *Economy* sebanyak 6 *sample*; nilai aktualnya *Automotive* dan prediksinya juga *Automotive* adalah sebanyak 6 *sample*. Evalusi kinerja model dari ke enam algoritma *machine learning* yang memiliki nilai akurasi pelatihan tertinggi dengan ekstraksi fitur Word2Vec disajikan pada Tabel 8. Nilai *accuracy*, *precision*, *recall* dan *f1-score* yang disajikan pada Tabel 14, berasal dari nil TP, TN, FP dan FN dari *confusion matrix* yang ada pada Gambar 8.

Tabel 14. Hasil Kinerja Evaluasi Pengujian Model dan Ekstraksi Fitur "Word2Vec"

Model	Label	*Precision*	*Recall*	*f1-score*	*Support*	Model	Label	*Precision*	*Recall*	*f1-score*	*Support*
NB ***Split* data 90:10**	*News*	0.31	0.67	0.43	24	**KNN** ***Split* data 80:20**	*News*	0.22	0.40	0.28	52
	Business	0.17	0.06	0.08	18		*Business*	0.21	0.27	0.24	37
	Sport	0.18	0.10	0.12	21		*Sport*	0.21	0.13	0.16	39
	Economy	0.31	0.45	0.37	20		*Economy*	0.13	0.08	0.10	37
	Automotive	0.67	0.12	0.20	17		*Automotive*	0.22	0.06	0.09	35
Akurasi Pengujian				0.30	100	**Akurasi Pengujian**				0.20	200
SVM ***Split* data 90:10**	*News*	0.40	0.25	0.31	24	**RF** ***Split* data 90:10**	*News*	0.45	0.42	0.43	24
	Business	0.33	0.39	0.36	18		*Business*	0.26	0.28	0.27	18
	Sport	0.47	0.38	0.42	21		*Sport*	0.34	0.33	0.36	21
	Economy	0.20	0.25	0.22	20		*Economy*	0.25	0.30	0.27	20
	Automotive	0.29	0.29	0.26	17		*Automotive*	0.33	0.35	0.5	17
Akurasi Pengujian				0.31	100	**Akurasi Pengujian**				0.34	100
DT **Split data 80:20**	*News*	0.35	0.23	0.28	52	**LR** ***Split* data 90:10**	News	0.29	0.29	0.29	24
	Business	0.17	0.22	0.19	37		*Business*	0.33	0.33	0.33	18
	Sport	0.22	0.21	0.21	39		*Sport*	0.23	0.14	0.18	21
	Economy	0.21	0.24	0.23	37		*Economy*	0.22	0.20	0.21	20
	Automotive	0.22	0.26	0.24	35		*Automotive*	0.11	0.18	0.14	17
Akurasi Pengujian				0.23	200	**Akurasi Pengujian**				0.25	100

Pengujian dari ke enam algoritma *machine learning* (i.e. NB, SVM, KNN, DT, RF, dan LR) dengan ekstraksi fitur Doc2Vec, disajikan pada Tabel 14. Dari pengujian diperoleh nilai *accuracy*, *precision*, *recall* dan *f1-score* dari tiap tiap model. Akurasi pengujian dan akurasi pelatihan mendekati sama. Berdasarkan AUC, ke enam algoritma *machine learning* (i.e. NB, SVM, DT, KNN, RF, dan LR) memiliki nilai akurasi pengujian dibawah 0.50 yang merupakan kategori *failure classification* untuk melakukan klasifikasi teks berdasarkan kategori berita dalam dataset tribunnews.com. Nilai *precision*, *recall* dan *f1-score* dari ke enam model juga memiliki nilai jauh di bawah 0.50. Hal ini menunjukkan bahwa algoritma *machine learning* dengan ekstraksi fitur Word2Vec tidak cocok untuk melakukan klasifikasi teks multilabel pada dataset tribunnews.com.

Penelitian yang sama melakukan komparasi terhadap *feature extraction* model (i.e CBOW, *Skip-gram*, *Glove*, dan *Hellinger-PCA*), GloVe adalah model terbaik dibandingkan dengan model lain [4]. Penelitian lain yang melakukan komparasi *word embedding* (i.e. LSA, Word2Vec dan Glove) pada segmentasi topik [1], menyatakan bahwa LSA, Word2Vec dan GloVe bergantung pada bahasa yang digunakan. Word2Vec menyajikan representasi vektor kata terbaik namun itu tergantung pada pilihan model. Penelitian terdahulu yang melakukan klasifikasi teks disajikan pada Tabel 15.

Tabel 15. Perbandingan Penelitian Sebelumnya

Author	*Feature Extraction*	*Classifier*	*Result*
Stein, Jaques, & Valiati [29]	Glove, Word2Vec, fastText	CNN, SVM, XGBoost	*Accuracy=0.89*
Shao dkk [30]	Word2Vec, BOW, Doc2Vec	SVM	*Accuracy=0.81*
Lilleberg dkk [12]	Word2Vec, TF-IDF	SVMperf	*Accuracy=0.89*
Gao dkk., [31]	Word2Vec+LDA	CNN	*Accuracy=0.83*
Yuan dkk [11]	Word2Vec+ TFIDF	Att-LSTM	*Accuracy=0.81*
Wang dkk., [32]	Label Embedding Attentive Model (LEAM)	CNN, RNN	*Accuracy=0.91*
Sun & Chen [33]	LDA+Word2vec+THFW+GCW	SVM	P=83.6; R=85.4; F1=84.4
Xu dkk., [34]	LDA+Skip gram	NB	0.81

4. KESIMPULAN

Model klasifikasi teks berdasarkan kategori berita pada tribunnews.com menggunakan empat *feature extraction* yang dikombinasikan dengan enam algoritma *machine learning*, dan perfoma model dievaluasi menggunakan *confusion matrix* dan kurva ROC berhasil dilakukan dan diperoleh *accuracy* yang bervariasi. *Feature extraction TF-IDF* dikombinsikan dengan NB, SVM, KNN, RF, dan LR pada *split* data 90:10 memperoleh akurasi di atas 80% dan berdasarkan kurva ROC merupakan kategori *good classification*. Sedangkan DT merupakan *poor classification* dengan *accuracy* dibawah 70%. Kombinasi BoW dan algoritma *machine learning* yang termasuk ke dalam *good classification* adalah NB, SVM, RF dan LR (*accuracy* di atas 80%); kategori *poor classification* adalah DT (*accuracy* dibawah 70%), sedangkan *failure classification* adalah KNN (*accuracy* dibawah 60%). Model NB+BoW lebih baik dari model yang lain. Untuk ekstraksi fitur Doc2Vec algoritma SVM merupakan good *classification* dan algoritma NB, DT, KNN, RF, dan LR termasuk *poor classification*. Dari enam algoritma *machine learning* yang memiliki akurasi pengujian tertinggi adalah SVM, artinya model SVM + Doc2Vec lebih baik dari model yang lain. Sedangkan untuk ekstraksi fitur Word2Vec, ke enam algoritma *machine learning*, memiliki nilai akurasi yang kurang dari 0.60 baik akurasi pelatihan, maupun akurasi pengujian. Berdasarkan AUC, ke enam algoritma dikategorikan sebagai *failure classification*. Hal ini menyatakan bahwa NB, SVM, DT, KNN, RF dan LR dengan ekstraksi fitur Word2Vec, telah gagal melakukan klasifikasi teks multilabel pada dataset tribunnews.com.

Kelemahan dari penelitian ini, *accuracy* model tidak satu pun kombinasi *feature extraction* dan algoritma *machine learning* memiliki nilai *accuracy* besar sama dengan 90% atau masuk kategori *excellent classification* berdasarkan kurva ROC. Penelitian selanjutnya, disarankan menambahkan algoritma yang dapat meningkatkan *accuracy* model dan menguji kembali kombinasi *feature extraction* dan algoritma *machine learning* pada dataset *public* seperti AGNews dan BBC News.

REFERENSI

[1] M. Naili, A. H. Chaibi, and H. H. Ben Ghezala, "Comparative study of word embedding methods in topic segmentation," *Procedia Computer Science*, vol. 112, pp. 340–349, 2017.

[2] F. K. Khattak, S. Jeblee, C. Pou-Prom, M. Abdalla, C. Meaney, and F. Rudzicz, "A survey of word embeddings for clinical text," *Journal of Biomedical Informatics: X*, vol. 4, p. 100057, 2019.

[3] A. Conneau, H. Schwenk, Y. L. Cun, and L. Barrault, "Very deep convolutional networks for text classification," *15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of Conference*, vol. 1, no. 2001, pp. 1107–1116, 2017.

[4] S. Bhoir, T. Ghorpade, and V. Mane, "Comparative analysis of different word embedding models," *International Conference on Advances in Computing, Communication and Control 2017, ICAC3 2017*, vol. 2018-Janua, pp. 1–4, 2018.

[5] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey," *Information (Switzerland)*, vol. 10, no. 4, pp. 1–68, 2019.

[6] A. Conneau, H. Schwenk, Y. Le Cun, and L. Barrault, "Very Deep Convolutional Neural Networks for Text Classification," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11727 LNCS, no. 2001, pp. 193–207, 2019.

[7] R. Wongso, F. A. Luwinda, B. C. Trisnajaya, O. Rusli, and Rudy, "News Article Text Classification in Indonesian Language," *Procedia Computer Science*, vol. 116, pp. 137–143, 2017.

[8] I. C. Irsan and M. L. Khodra, "Hierarchical multi-label news article classification with distributed semantic model based features," *International Journal of Advances in Intelligent Informatics*, vol. 5, no. 1, pp. 40–47, 2019.

[9] A. Onan, S. Korukolu, and H. Bulut, "Ensemble of keyword extraction methods and classifiers in text classification," *Expert Systems with Applications*, vol. 57, pp. 232–247, 2016.

[10] Y. Goldberg and O. Levy, "word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method," no. 2, pp. 1–5, 2014.

[11] H. Yuan, Y. Wang, X. Feng, and S. Sun, "Sentiment analysis based on weighted word2vec and ATT-LSTM," *ACM International Conference Proceeding Series*, pp. 420–424, 2018.

[12] J. Lilleberg, Y. Zhu, and Y. Zhang, "Support vector machines and Word2vec for text classification with semantic features," *Proceedings of 2015 IEEE 14th International Conference on Cognitive Informatics and Cognitive Computing, ICCI*CC 2015*, pp. 136–140, 2015.

[13] R. G. Rossi, A. D. A. Lopes, and S. O. Rezende, "Optimization and label propagation in bipartite heterogeneous networks to improve transductive classification of texts," *Information Processing and Management*, vol. 52, no. 2, pp. 217–257, 2016.

[14] Z. Liu, Y. Lin, and M. Sun, *Representation Learning for Natural Language Processing*, 2020.

[15] B. Y. Pratama and R. Sarno, "Personality classification based on Twitter text using Naive Bayes, KNN and SVM," *Proceedings of 2015 International Conference on Data and Software Engineering, ICODSE 2015*, pp. 170–174, 2016.

[16] M. Azam, T. Ahmed, F. Sabah, and M. I. Hussain, "Feature Extraction based Text Classification using K-Nearest Neighbor Algorithm," *IJCSNS International Journal of Computer Science and Network Security*, vol. 18, no. 12, pp. 95–101, 2018.

[17] S. Xu, "Bayesian Naïve Bayes classifiers to text classification," *Journal of Information Science*, vol. 44, no. 1, pp. 48–59, 2018.

[18] L. Jiang, C. Li, S. Wang, and L. Zhang, "Deep feature weighting for naive Bayes and its application to text classification," *Engineering Applications of Artificial Intelligence*, vol. 52, pp. 26–39, 2016.

[19] Y. Cahyono, "Analisis Sentiment pada Sosial Media Twitter Menggunakan Nave Bayes Classifier dengan Feature Selection Particle Swarm Optimization dan Term Frequency," *Jurnal Informatika Universitas Pamulang*, vol. 2, no. 1, p. 14, 2017.

[20] M. Fanjin, H. Ling, T. Jing, and X. Wang, "The research of semantic kernel in SVM for Chinese text classification," *ACM International Conference Proceeding Series*, vol. Part F1318, no. 319, 2017.

[21] W. A. Luqyana, I. Cholissodin, and R. S. Perdana, "Analisis Sentimen Cyberbullying Pada Komentar Instagram dengan Metode Klasifikasi Support Vector Machine," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, vol. 2, no. 11, pp. 4704–4713, 2018.

[22] C. Satria and A. Anggrawan, "Aplikasi K-Means berbasis Web untuk Klasifikasi Kelas Unggulan," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 21, no. 1, pp. 111–124, 2021.

[23] A. Muhammad and S. Defit, "Analyzing the use of Social Media by Fashion Designers with K-Means and C45," vol. 21, no. 2, pp. 463–476, 2022.

[24] K. I. Gunawan and J. Santoso, "Multilabel Text Classification Menggunakan SVM dan Doc2Vec Classification Pada Dokumen Berita Bahasa Indonesia," *Journal of Information System,Graphics, Hospitality and Technology*, vol. 3, no. 01, pp. 29–38, 2021.

[25] M. Gao, T. Li, and P. Huang, *Text classification research based on improved word2vec and CNN*. Springer International Publishing, 2019, vol. 11434 LNCS.

[26] M. K. Anam, B. N. Pikir, and M. B. Firdaus, "Penerapan Na ve Bayes Classifier, K-Nearest Neighbor (KNN) dan Decision Tree untuk Menganalisis Sentimen pada Interaksi Netizen danPemeritah," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 21, no. 1, pp. 139–150, 2021.

[27] D. R. Java, W. Wijaya, J. Hendry, and B. Sumanto, "Seleksi Fitur Terhadap Performa Kinerja Sistem E-Nose untuk Klasifikasi Aroma Kopi Gayo Features Selection on E-Nose System Performance for Classification of Gayo Coffee Aroma," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 2, 2022.

[28] F. Gorunescu, *Data Mining: Concepts, models and techniques. Vol (12)*. Springer Science & Business Media, 2011.

[29] R. A. Stein, P. A. Jaques, and J. F. Valiati, "An analysis of hierarchical text classification using word embeddings," *Information Sciences*, vol. 471, pp. 216–232, 2019.

[30] Y. Shao, S. Taylor, N. Marshall, C. Morioka, and Q. Zeng-Treitler, "Clinical Text Classification with Word Embedding Features vs. Bag-of-Words Features," *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*, pp. 2874–2878, 2019.

[31] X. Wang, D. Gao, G. Zhang, X. Zhang, Q. Li, Q. Gao, R. Chen, S. Xu, L. Huang, Y. Zhang, L. Lin, C. Zhong, X. Chen, G. Sun, Y. Song, X. Yang, L. Hao, H. Yang, L. Yang, and N. Yang, "Exposure to multiple metals in early pregnancy and gestational diabetes mellitus: A prospective cohort study," *Environment International*, vol. 135, p. 105370, 2020.

[32] G. Wang, C. Li, W. Wang, Y. Zhang, D. Shen, X. Zhang, R. Henao, and L. Carin, "Joint embedding of words and labels for text classification," *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, vol. 1, pp. 2321–2331, 2018.

[33] F. Sun and H. Chen, "Feature extension for Chinese short text classification based on LDA and Word2vec," *Proceedings of the 13th IEEE Conference on Industrial Electronics and Applications, ICIEA 2018*, no. 1, pp. 1189–1194, 2018.

[34] H. Xu, A. Kotov, M. Dong, A. I. Carcone, D. Zhu, and S. Naar-King, "Text classification with topic-based word embedding and Convolutional Neural Networks," *ACM-BCB 2016 - 7th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics*, no. April 2019, pp. 88–97, 2016.