INSYST

Journal of Intelligent System and Computation

p-ISSN:2621-9220
e-ISSN:2722-1962

Volume 4 Nomor 2, Oktober 2022

Published By **Lembaga Penelitan dan Pengabdian Masyarakat (LPPM)**
Institut Sains dan Teknologi Terpadu Surabaya (ISTTS)
formerly **Sekolah Tinggi Teknik Surabaya (STTS)**

Managed By
Departement of Informatics
Institut Sains dan Teknologi Terpadu Surabaya (ISTTS)

INSYST

Journal of Intelligent System and Computation

Volume 04 Nomor 02 Oktober 2022

INSYST

Journal of Intelligent System and Computation

Volume 04 Nomor 02 Oktober 2022

Reviewer:

INSYST

Journal of Intelligent System and Computation

Volume 04 Nomor 02 Oktober 2022

Reviewer:

Windra Swastika, Ph.D

Universitas Ma Chung, Indonesia

Romy Budhi Widodo, Dr.Eng.

Universitas Ma Chung, Indonesia

INSYST

Journal of Intelligent System and Computation

Volume 04 Nomor 02 Oktober 2022

Daftar Isi

Pengenalan Makanan Tradisional Indonesia Beserta Bahan-bahannya dengan Memanfaatkan DCNN Transfer Learning

Citra Mahaputri[1], Yosi Kristian[1], Endang Setyati[1]
[1]Departemen Informatika, Fakultas Sains dan Teknologi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Indonesia

Corresponding author: Citra Mahaputri (e-mail: citramahaputri@gmail.com).

ABSTRACT Food recognition is the first step to assessing one's diet. In the introduction of food and its ingredients, it is felt that there is a lack of dissemination of photos of traditional Indonesian food, so researchers are encouraged to conduct research on the recognition of traditional Indonesian food. The researcher made a classification of food images whose input is an image of traditional Indonesian food. Feature extraction of food images is difficult to classify because food images vary in appearance, including texture, colour, shape and other visual characteristics. This research examines the use of Deep Convolutional Neural Network (DCNN) models EfficientNetB6 and EfficientNetV2M for the recognition of traditional Indonesian food and its ingredients. DCNN is a method commonly used to detect complex images. Researchers manually collected 1,202 different images of traditional Indonesian food. It consists of 20 types of traditional Indonesian food. Each type of food has 50-80 food images. The data used for food classification tests is 241 food image data outside the data used for training and get 83.82% accuracy for the EfficientNetV2M model and 80.08% for the EfficientNetB6 model. Then in the testing process in predicting the food ingredients seen in the image on average get 64% for the EfficienNetV2M model and 59% for the EfficeintNetB6 model. Based on the research results, it shows that the DCNN method with the EfficientNetV2M model can achieve the best performance of the EfficientNetB6 model.

KEYWORDS Deep Convolution Neural Network, EfficientNetB6, EfficientNetV2M, Image Classification

ABSTRAK Pengenalan makanan adalah langkah awal untuk melakukan penilaian diet seseorang. Dalam pengenalan makanan beserta bahan-bahannya, dirasakan kurang diseminasi foto-foto makanan tradisional Indonesia, sehingga peneliti terdorong untuk melakukan penelitian mengenai pengenalan makanan tradisional Indonesia. Peneliti membuat klasifikasi citra makanan yang inputannya merupakan citra makanan tradisional Indonesia. Ekstraksi fitur citra makanan sulit untuk diklasifikasikan karena citra makanan beraneka ragam penampilannya, termasuk tekstur, warna, bentuk dan karakteristik visual lainnya. Penelitian ini meneliti pemanfaatan *Deep Convolutional Neural Network* (DCNN) model *EfficientNetB6* dan *EfficientNetV2M* untuk pengenalan makanan tradisional Indonesia beserta bahan-bahannya. DCNN merupakan metode yang biasa digunakan untuk mendeteksi citra yang komplek. Peneliti mengumpulkan citra makanan tradisional Indonesia secara manual sebanyak 1.202 citra makanan yang berbeda. Terdiri dari 20 jenis makanan tradisional Indonesia. Masing-masing jenis makanan terdapat 50-80 gambar makanan. Data yang digunakan untuk uji klasifikasi makanan adalah 241 data citra makanan di luar data yang digunakan untuk *training* dan mendapatkan akurasi 83,82% untuk model *EfficientNetV2M* dan 80,08% untuk model *EfficientNetB6.* Kemudian pada proses pengujian dalam memprediksi bahan-bahan makanan yang terlihat pada gambar rata-rata mendapatkan 64% untuk model *EfficienNetV2M* dan 59% untuk model *EfficeintNetB6*. Berdasarkan hasil penelitian menunjukkan bahwa metode DCNN dengan model *EfficientNetV2M* dapat mencapai performa terbaik dari model *EfficientNetB6.*

KATA KUNCI *Deep Convolution Neural Network, EfficientNetB6*, *EfficientNetV2M,* Klasifikasi Makanan

I. PENDAHULUAN

Indonesia memiliki aneka ragam tradisi, budaya dan aneka ragam kuliner yang sangat menarik untuk di coba. Gaya hidup masyarakat mempengaruhi pemenuhan kebutuhan kalori dan sudah menjadi masalah yang umum di hadapi pada masa sekarang ini. Hal utama yang menjadi penyebab permasalahan ini adalah jenis makanan yang dimakan oleh manusia.

Pola makan yang tidak sehat berdampak besar pada nutrisi yang dibutuhkan tubuh. Makanan cepat saji dan makanan tinggi gula dapat meningkatkan risiko terhadap kesehatan [1],[2],[3]. Penerapan panduan nutrisi seimbang menggunakan metode *chaining backward* memberikan informasi tentang panduan nutrisi seimbang dan kalkulator kalori. Metode *chaining backward* digunakan untuk menghitung nilai gizi dan kalori pengguna aplikasi berdasarkan umur, makanan yang dimakan, tinggi dan berat badan [4].

Penelitian dalam pengenalan makanan merupakan perkembangan di masa sekarang. Sebagian besar peneliti mendapat tantangan tidak hanya mendeteksi nama makanan tetapi juga menghitung kalori di dalamnya [5]. Peneliti melakukan penelitian mengenai pengenalan makanan dan estimasi nutrisi dengan menggunakan *Machine Learning* dan menyiapkan dua model sistem dalam penelitiannya. Model pertama adalah model *text mining* untuk mengumpulkan informasi makanan di lebih dari 500 *website* dengan menggunakan *crawling* dan *scrapy*. Tahap selanjutnya, informasi berupa kata-kata diekstraksi dengan dipisahkan dengan menggunakan library *Python HTML Parser* untuk mendapatkan informasi penting mengenai nama masakan, bumbu – bumbu dan komposisi masakan tersebut. Peneliti membuat bank data mengenai makanan dengan resep/bumbu makanannya dengan menggunakan *word2vec*.

Keuntungan dari metode yang ada dalam penelitian [5] adalah menggabungkan gambar dengan informasi yang terkandung di dalamnya untuk menggabungkan dan memunculkan data informasi kalori dan nutrisi dari bumbu dan komposisi masakan. Penelitian ini memiliki akurasi sebesar 85%. Kelemahan dari penelitian ini adalah *dataset* dan variasi makanan yang digunakan penulis masih kurang banyak. Kemudian justifikasi kebenaran mengenai jumlah kalori dan nutrisi dalam makanan belum terverifikasi dengan baik oleh ahlinya.

Kontribusi untuk penelitian ini adalah :

1. Merancang sebuah sistem klasifikasi makanan yang dapat memprediksi makanan tradisional Indonesia yang inputnya merupakan citra dari makanan tradisional Indonesia dengan memanfaatkan metode *Transfer Learning* pada DCNN dengan model *EfficientNetB6* dan *EfficientNetV2M* untuk memprediksi ini makanan apa.
2. Merancang sebuah sistem yang dapat memprediksi bahan - bahan makanan pada makanan tradisional Indonesia dengan memanfaatkan metode *Transfer Learning* pada DCNN dengan model *EfficientNetB6* dan *EfficientNetV2M*.

II. PENELITIAN YANG RELEVAN

Pada tahun 2019, [6] melakukan penelitian yang terkait dengan pengenalan makanan Cina dengan menggunakan metode CNN. Peneliti membangun model penelitian dengan menggunakan perbandingan dan korelasi antara arsitektur CNN yang diusulkan dan model *BoF* (*Bag-of Feature*). Keunggulannya terletak pada kombinasi fitur dan klasifikasi yang terdiri dari lapisan yang terhubung penuh. Tujuan menggunakan *layer* yang terhubung penuh adalah untuk menggabungkan fitur tingkat tinggi yang diekstraksi dari fase sebelumnya untuk mengklasifikasikan gambar input ke dalam kelas yang sesuai seperti yang ditentukan oleh *dataset* pelatihan. Penggabungan kombinasi fitur dengan klasifikasi menggunakan *layer* terhubung mencegah hilangnya data pada proses *training*, sehingga pada saat pengujian dilakukan, sistem mencapai akurasi yang baik. Data yang digunakan sebanyak 8734 gambar dari 25 jenis makanan yang berbeda di China. Metode CNN memiliki akurasi untuk top-1 sebesar 97,12% dan tingkat akurasi untuk top-5 sebesar 99,86%. Kesimpulan tersebut cukup kuat untuk mendukung keunggulan yang diklaim karena kombinasi kombinasi fitur dan klasifikasi menggunakan *layer* terhubung mencegah hilangnya data pada proses *training*, sehingga pada saat pengujian dilakukan sistem mencapai akurasi yang baik. Kekurangan dari penelitian ini adalah *dataset* gambar terbatas pada masakan Cina. Peneliti menyarankan untuk penelitian selanjutnya dapat ditambahkan daftar makanan dari negara lain.

Pada tahun 2020, [7] melakukan penelitian mengenai pengenalan dalam 13 kategori makanan di Vietnam [7]. Peneliti mencoba membandingkan 2 metode utama yaitu *Hand-Craft Feature* dan CNN. *Hand-Craft Feature* adalah teknik pengenalan objek yang mengekstrak informasi gambar seperti warna, tekstur, dan bentuk. Metode *Hand-Craft Feature* yang digunakan peneliti adalah *Support Vector Machines* (*SVMs*) dan *Extreme Gradient Boosting (XGBoost).* Metode berbasis CNN yang dipakai pada penelitian ini antara lain *AlexNet, GoogleNet, ResNet50, ResNet101v2*, dan *InceptionResnetv2.* Peneliti menggunakan metode optimasi adam untuk pembobotan setiap *layer* dari metode CNN. Peneliti mendapatkan akurasi tertinggi 1,3,dan 5 dengan menggunakan 8093 gambar dari 13 jenis makanan Vietnam yang berbeda. *InceptionResnetv2* memiliki hasil akurasi yang tinggi dibandingkan dengan metode lain. Terutama pada *layer* 3 dan 5. *InceptionResnetv2* seharusnya memiliki akurasi yang tinggi karena memiliki jumlah *layer* dan parameter paling banyak dibandingkan metode lainnya. Peneliti menyarankan untuk menambahkan jumlah kategori gambar dan menambahkan justifikasi mengenai nutrisi dan kalori makanan. Penelitian ini terbatas pada komputer dan nantinya dapat diimplementasikan pada *mobile phone.*

Pada penelitian selanjutnya, [8] bertujuan untuk mendeteksi makanan Indonesia dengan menggunakan metode CNN melakukan normalisasi (*cropping, wrapping, resizing*) data masukan. Tahap selanjutnya, peneliti mengubah ukuran citra menjadi 128 x 128 piksel dan mengubah citra menjadi *grayscale* untuk proses pelatihan. Metode CNN yang digunakan adalah *forward-propagation* dan *back-propagation*. Dalam proses *training*, peneliti menggunakan sekitar 10 ribu data dengan 10 kelas, setiap kelas makanan memiliki 1000 data. Untuk pengujian, peneliti menggunakan 500 gambar, dimana setiap kelas menggunakan 50 gambar. Hasil penelitian menunjukkan keakuratan pengenalan makanan sebesar 88% menggunakan metode CNN. Kesimpulan ini mendukung hasil penelitian, karena gambar dinormalisasi sebelum diproses. Semakin optimal penggunaan data *training* maka semakin tinggi akurasi yang dihasilkan.

Pada penelitian sebelumnya telah meneliti pemanfaatan DCNN untuk mengenali budaya Indonesia berupa Ukiran Jepara [9] sedangkan pada [10] meneliti pemanfaatan DCNN untuk mengenali nyeri dari wajah bayi. Penelitian tentang pengenalan makanan lainnya dapat ditemukan di [11], [12], [13].

Berdasarkan penelitian sebelumnya hasil yang didapatkan cukup memuaskan, namun ekstraksi fitur tidak berjalan dengan baik karena mengandalkan fitur *BoF (Bag-of Feature)* dan *Hand-Craft Feature* yang sering menjadi kendala. Oleh karena itu, dalam penelitian ini menerapkan ekstraksi fitur dengan *feature learning*. Bagian berikut menjelaskan tentang *Convolutional Neural Network* (CNN) dan *EfficientNet* yang digunakan dalam penelitian ini.

III. *Convolutional Neural Network* dan *EfficientNet*

Dalam penelitian ini, peneliti menggunakan metode *Convolutional Neural Network* (CNN*)* yang merupakan pengembangan dari *Deep Convolutional Neural Network* (DCNN), jenis *neural network* yang biasa digunakan untuk data terkait gambar. CNN bisa digunakan untuk mendeteksi dan mengenali obyek dalam gambar. Fitur utama CNN adalah model/ arsitektur yang dapat melihat informasi prediktif suatu objek ketika diposisikan dimana saja pada *input*. CNN terdiri dari banyak *neuron* yang memiliki *weight, bias* dan *activation function. Convolutional Neural Network* (CNN) terdapat 2 bagian utama, yaitu *feature learning* dan *classification.* Berikut adalah penjelasan dari kedua bagian utama tersebut. *Convolutional Neural Network* (CNN) terdapat 2 bagian utama, yaitu *feature learning* dan *classification.* 2 bagian utama dijelaskan dibawah ini

A. Feature Learning

Feature learning adalah melakukan proses "*encoding*" suatu citra menjadi fitur dalam bentuk numerik untuk merepresentasikan citra tersebut. Feature learning terdiri dari beberapa layer yang bekerja sama untuk mendapatkan gambar. Penjelasan dari setiap *layer* adalah sebagai berikut:

1) *Convolution Layer*

Convolution Layer pada CNN menghasilkan citra baru yang mewakili fitur dari gambar input. Dalam proses ini, *Convolution Layer* pada CNN menggunakan filter pada setiap gambar masukan. Proses ini menggunakan filter dengan tinggi, lebar, dan ketebalan tertentu. Filter ini diinisialisasi dengan nilai tertentu dan nilai filter ini yang menjadi parameter yang akan diperbarui dalam proses *learning* [14]. Konvolusi adalah istilah matematika yang berarti berulang kali menerapkan satu fungsi ke output fungsi lain. Dalam pemrosesan gambar, konvolusi berarti menerapkan kernel ke semua kemungkinan skema *offset* .

2) *Activation Layer*

Activation Layer adalah *layer* yang fungsi aktivasinya *feature map* yang dihasilkan dari *convolution layer*. Fungsi aktivasi membantu mengubah nilai pada *feature map* dalam rentang tertentu sesuai dengan fungsi aktivasi yang digunakan. Tujuannya adalah untuk meneruskan nilai ke *layer* berikutnya yang menjelaskan fitur utama dari gambar yang masuk. Peneliti biasanya menggunakan fungsi aktivasi *reLU* yang lebih fungsional. Fitur aktivasi *reLU* memungkinkan nilai output neuron direpresentasikan sebagai 0 ketika input negatif. Jika masukannyandari fungsi ativiasi adalah positif, keluaran dari neuron menjadi masukan dari aktivasi itu sendiri.

3) *Pooling layer*

Polling layer biasanya muncul setelah *convolution layer* dan *activation layer*. Pada dasarnya *pooling layer* terdiri dari filter dengan ukuran tertentu yang bergerak di semua area *feature map*. *Pooling layer* yang umum digunakan adalah *Max Pooling* dan *Average Pooling*. Tujuan penggunaan *pooling layer* adalah untuk memperkecil ukuran *feature map (down sampling)*, yang meningkatkan kecepatan komputasi karena lebih sedikit parameter yang perlu diperbarui dan dapat mengatasi *overfitting* [15]. *Overfitting* terjadi ketika sampel data *training* terlalu acak dan *loss* berkurang sementara *val_loss* tetap sama atau meningkat. Hal terpenting saat membuat model CNN adalah memilih berbagai *pooling layer* [16].

B. Classification

Tujuan dari proses *classification* adalah untuk mengklasifikasikan setiap *neuron* yang diekstrak dalam proses *feature learning.* Bagian ini terdiri dari beberapa layer yang saling berhubungan. Berikut adalah penjelasan dari masing-masing fungsi yang termasuk dalam *classification* tersebut.

1) *Flatten*

Feature learning menghasilkan *feature map* dari masih berbentuk *multidimensional array*, sedangkan agar sesuai dengan *fully-connected layer,* data harus dalam bentuk vektor, sehingga membutuhkan fungsi yang disebut *Flatten*. Fungsi *flatten* adalah untuk membentuk kembali (*reshape*) *feature map* menjadi vektor sehingga dapat digunakan sebagai *input* dari *fully-connected layer*.

2) *Fully Connected Layer*

Fully Connected Layer umum digunakan dalam penerapan *multi-layer perceptron* dan dimaksudkan untuk mengubah dimensi data sehingga data dapat diklasifikasikan secara linear. *Layer* ini memiliki *hidden layer*, *activation function*, *output layer*, dan *loss function.* Setiap neuron pada *convolution layer* harus diubah menjadi data satu dimensi sebelum dapat dimasukkan ke dalam *fully-connected layer*. Karena jika data kehilangan informasi spasial dan tidak dapat dibalik, sedangkan *fully-connected layer* hanya dapat dilakukan di ujung jaringan. *Convolution layer* dengan ukuran kernel 1 x 1 melakukan fungsi yang sama dengan *fully-connected layer* tetapi tetap mempertahankan karakter spasial data.

Dalam penelitian pengenalan makanan tradisional Indonesia beserta bahan - bahannya peneliti menggunakan *framework* untuk mempermudah pembuatan program *deep learning*. Peneliti menggunakan *Keras Applications*. *Keras Applications* merupakan salah satu modul *library* yang menyediakan arsitektur untuk berbagai model *deep learning*. Model deteksi gambar *Keras Applications* juga dapat digunakan untuk mengekstrak fitur dari citra. *Keras* adalah antarmuka yang memudahkan pengguna dalam pemrograman dan semua komputasi model dilakukan oleh *library* lain yaitu *TensorFlow* atau *Theano.*

Peneliti menggunakan sebuah kerangka kerja/*framework* yang bernama *tensorflow*. Penggunaan *tensorflow* pada penelitian ini untuk mempermudah dan mempercepat proses *training* dan uji coba penelitian. *Tensorflow* sudah banyak digunakan untuk penelitian berbasis *deep learning-convolutional neural network,* [17], [18].

Pada penelitian ini, peneliti menggunakan *transfer learning* untuk pengenalan jenis makanan dan pengenalan bahan - bahan yang terlihat pada gambar makanan. *Transfer learning* adalah alat yang digunakan untuk mentransfer pengetahuan dari *domain* asal ke *domain* tujuan/target [19], [20].

EfficientNet adalah arsitektur dari *Convolutional neural network* (CNN) yang di dasarkan pada 3 dimensi penskalaan sederhana dan sangat efektif, yaitu *depth* (kedalaman), *width* (lebar) dan *resolution* (resolusi). Dimensi penskalaan pada *EfficientNet* merupakan penskalaan dengan koefisien yang tetap. Secara umum *EfficientNet* dapat mencapai nilai akurasi yang tinggi dan memiliki tingkat performa yang lebih baik dari arsitektur CNN yang lain dan dapat mengurangi ukuran parameter dan FLOPS yang optimal [21]. *EfficientNetB6* adalah arsitektur *Convolutional Neural Network* (CNN) yang dibuat berdasarkan ide dasar CNN menjadi arsitektur *Neural Network* dengan sumber daya tetap dan kemudian ditingkatkan untuk mendapatkan akurasi tinggi. Dalam prakteknya, membangun arsitektur CNN sesuai dengan pola penskalaan tertentu dan mengetahui bahwa menyeimbangkan kedalaman, lebar, dan resolusi jaringan dengan hati-hati dapat menghasilkan kinerja yang lebih baik. Sedangkan *EfficientNetV2M* merupakan salah satu jenis *convolutional neural network* dengan kecepatan *training* yang lebih cepat dan efisiensi parameter yang lebih baik dari model sebelumnya.

C. *Confusion Matrix*

Confusion Matrix adalah metode yang digunakan untuk menampilkan dan membandingkan nilai aktual dengan nilai prediksi dari model yang dapat digunakan untuk memberikan ukuran evaluasi seperti akurasi, *Precission*, *Recall*, *F1Score*. Akurasi adalah alat ukur untuk menentukan tingkat kesesuaian hasil pengukuran dengan nilai sebenarnya. Akurasi dituliskan pada (1). *F1Score* adalah ukuran level *classifier* dapat menilai sebuah kelas. *F1Score* dapat digunakan untuk mengukur kinerja *multiclass* dengan mengambil rata-rata tertimbang (*weighted average*) hasil *F1Score* untuk semua kelas seperti yang ditunjukkan pada (2). Bobot di sini adalah jumlah data untuk mendukung (support) setiap kelas. *Precision* adalah tingkat ketelitian antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem, seperti ditunjukkan pada (3). *Recall* adalah tingkat keberhasilan sistem dalam mengambil informasi. Dengan persamaan (4) nilai *Recall* dapat diketahui. Ada 4 nilai yang dihasilkan oleh *Confusion Matrix* diantaranya: Nilai *True Negative* (TN) adalah data yang diklasifikasikan dengan benar sebagai keluaran negatif atau salah. *True Positive* (TP) adalah data yang diklasifikasikan dengan benar sebagai output positif atau benar. *False Positive* (FP) adalah data yang diklasifikasikan apakah keluarannya positif atau benar. *False Negative* (FN) adalah data yang diklasifikasi dengan kurang tepat sehingga keluarannya negatif atau salah [22].

$$Akurasi = \frac{True\ Positive}{Total\ Data} \quad (1)$$

$$F1Score = \frac{2*Precission*Recall}{Precission+Recall} \quad (2)$$

$$Precission = \frac{True\ Positive}{True\ Positive+False\ Positive} \quad (3)$$

$$Recall = \frac{True\ Positive}{True\ Positive+False\ Negative} \quad (4)$$

IV. *METODE*

A. *Dataset*

Dalam penelitian ini, peneliti memilih makanan tradisional Indonesia yang dijadikan objek penelitian. Gambar makanan tradisional Indonesia dikumpulkan secara manual sebanyak 1.202, gambar makanan yang terdiri dari 20 jenis makanan tradisional Indonesia yang berbeda diambil dari situs *web* dan foto selama 3 bulan berturut-turut. Pada masing-masing kelas makanan terdapat 50-80 gambar

makanan yang sudah dikumpulkan menjadi *dataset* yang siap untuk dilakukan pengujian klasifikasi makanan dan memprediksi bahan-bahannya. Setelah itu peneliti melakukan pelabelan yang sesuai dengan kelas makanan. 80% dari 1.202 yaitu 961 data citra makanan digunakan untuk data *training* dan 20% yaitu 241 data citra makanan digunakan untuk data *testing*.

B. *Image Preprocessing*

Dalam proses pengumpulan data gambar makanan, peneliti menggunakan gambar makanan dengan kriteria berikut ini :

1. Gambar tampak atas
2. Gambar disimpan dalam bentuk JPG
3. Gambar berukuran 224 x 224
4. Gambar harus terdiri dari makanan (minuman, alat makan, hiasan, pot bunga dan lainnya tidak di perbolehkan)
5. Gambar makanan berwarna
6. Jenis gambar makanan terdiri dari 20 kelas makanan tradisional Indonesia yaitu Ayam Lodho, Gado-gado, Gule Kambing, Krengsengan, Lodeh, Lontong Kikil, Opor Ayam, Oseng-oseng Kerang, Oseng-oseng Udang, Rawon, Rujak Cingur, Sate Madura, Sayur Asem, Sayur Bayam, Sayur Bobor, Sop Ayam, Sop Daging, Soto Lamongan, Tahu Campur, Tahu Tek.

Tahap pertama, citra makanan yang telah terkumpul diolah kemudian masuk ke dalam tahap *preprocessing*. Pada proses *wrapping,* bertujuan untuk menentukan tepi citra, selanjutnya melakukan proses *cropping* objek pada citra sehingga sistem dapat fokus untuk mengidentifikasi objek utama citra makanan. Tujuan dilakukan *cropping* adalah untuk membuat citra makanan lebih jelas dan sama untuk semua gambar. *Cropping* yaitu proses menghilangkan bagian gambar yang tidak di perlukan. Proses ini biasanya meliputi penghapusan sebagian tepi citra untuk menghilangkan sampah asing dari citra, meningkatkan frekuensi gambar dan mengubah rasio [14]. Proses ini dilakukan secara manual dengan menggunakan software. Selanjutnya adalah pelabelan dan pengelompokan gambar. Gambar-gambar yang sudah dipilah dan ukurannya sama, dikelompokkan dengan jenis kelas yang sama dalam satu *folder* dan di beri nama menurut 20 kelas makanan tradisonal Indonesia untuk mempermudah proses *training.* Proses pengolahan data gambar makanan tradisional Indonesia dimulai dari mengubah ukuran citra makanan menjadi ukuran 224 x 224, kemudian mengubah menjadi *grayscale* agar mudah diproses pada tahap *training.* Kemudian citra makanan diolah dengan menggunakan *deep learning* untuk proses *training*. Pada tahap *training* dimulai dengan mengubah citra menjadi vektor.

C. *Training*

Pada proses *training* bertujuan untuk menghasilkan akurasi yang tinggi dari klasifikasi citra makanan. Tahap *training* ini terdiri dari proses *feedforward* dan proses *backpropagation.* Untuk memulai proses *feedforward*, membutuhkan jumlah dan ukuran layer yang dibuat, ukuran subsampling dan citra vektor. Kemudian proses *feedforward* pada citra vektor akan melalui konvolusi dan *Max Pooling* untuk memperkecil ukuran gambar dan memperbanyak neuron. Pada proses *training* peneliti menggunakan data sebanyak 961 data citra makanan. Peneliti melakukan *Training* dengan membandingkan model DCNN *EfficientNetB6* dan *EfficientNetV2M* dengan *epoch* 100, *batchsize* 50. Peneliti mendapatkan nilai akurasi 0,9834 dengan nilai kerugian 0,0521 untuk model *EfficientNetB6*, dan model *EfficientNetV2M* mendapatkan akurasi 0,9979 dengan nilai kerugian 0,0062.

V. HASIL

A. Implementasi

Penelitian ini memanfaatkan metode *Deep Convolutional Neural Network (DCNN)* yang mampu mengklasifikasikan gambar makanan tradisional Indonesia dan memprediksi bahan-bahan makanan pada makanan tersebut dengan menggunakan model *EfficienNetB6* dan *EfficienNetV2M*. Dalam penerapannya, ada beberapa langkah untuk memprediksi proses klasifikasi gambar makanan, yang pertama, gambar makanan dimasukkan dalam proses konvolusi dengan, *input* (224, (3, 3)) untuk mengurangi dimensi setiap gambar *input* makanan tetapi masih menyimpan informasi penting dari gambar. Proses "*Flattening*" dan "*Fully-connected layer*" untuk mengklasifikasikan fitur yang di peroleh dalam proses sebelumnya pada semua kelas. Gambar 1 adalah hasil klasifikasi makanan dengan metode DCNN yang menghasilkan klasifikasi Sate Madura dan Ayam Lodho.

GAMBAR 1. Hasil Klasifikasi Citra Makanan

B. Pengujian

Dalam penelitian ini telah dilakukan penerapan metode *Convolutional Neural Network* (CNN), metode ini dapat menghasilkan klasifikasi citra makanan. Pada pelaksanaanya, sistem dalam melakukan klasifikasi citra makanan cukup baik, dengan prediksi kesalahan sistem sebesar 19,92% dalam citra makanan. Dalam menguji keakuratan sistem, peneliti melakukan pengujian pada 20 kelas makanan dan menggunakan 241 citra makanan baru

dari setiap kelas pengujian diluar gambar yang digunakan untuk *training*.

Berikut ini adalah hasil pengujian dengan model *EfficentNetB6* yang sudah di lakukan oleh peneliti:

TABEL I.
AKURASI HASIL PENGUJIAN CITRA MAKANAN DENGAN *EFFICIENTNETB6*

Nama Makanan	BENAR		SALAH		Total Gambar Pengujian
	Jumlah Gambar	Persentase (%)	Jumlah Gambar	Persentase (%)	
Ayam Lodho	13	72,22	5	28	18
Gado - Gado	12	80,00	3	20	15
Gule Kambing	13	86,66	2	13	15
Krengsengan	17	94,44	1	6	18
Lodeh	14	87,50	2	12	16
Lontong Kikil	6	54,54	5	45	11
Opor Ayam	2	33,33	4	67	6
Oseng – Oseng Kerang	8	80,00	2	20	10
Oseng – Oseng Udang	14	93,33	1	7	15
Rawon	10	90,90	1	9	11
Rujak Cingur	8	66,66	4	33	12
Sate Madura	7	100,00	0	0	7
Sayur Asem	12	85,71	2	14	14
Sayur Bayam	6	54,54	5	14	11
Sayur Bobor	7	87,50	1	12	8
Sop Ayam	14	82,35	3	18	17
Sop Daging	7	63,63	4	36	11
Soto Lamongan	10	100,00	0	0	10
Tahu Campur	8	88,88	1	11	9
Tahu Tek	5	71,42	2	29	7
Rata-rata	193	80,08	48	19,92	241

Berdasarkan hasil pengujian klasifikasi citra makanan dengan model *EfficientNetB6* pada Tabel I mendapatkan hasil pengujian sistem dengan akurasi 80,08% dan kesalahan rata-rata 19,92%.

Dibawah ini adalah hasil pengujian menggunakan metode *CNN* dengan model *EfficentNetV2M* yang sudah di lakukan oleh peneliti:

TABEL II.
AKURASI HASIL PENGUJIAN CITRA MAKANAN DENGAN *EFFICINTNETV2M*

Nama Makanan	BENAR		SALAH		Total Gambar Pengujian
	Jumlah Gambar	Persentase (%)	Jumlah Gambar	Presentase (%)	
Ayam Lodho	14	77,78	4	22,22	18
Gado - Gado	14	93,33	1	6,67	15
Gule Kambing	13	86,67	2	13,33	15
Krengsengan	17	94,44	1	5,56	18
Lodeh	13	81,25	3	18,75	16
Lontong Kikil	8	72,73	3	27,27	11
Opor Ayam	4	66,67	2	33,33	6
Oseng – Oseng Kerang	9	90,00	1	10,00	10
Oseng – Oseng Udang	14	93,33	1	6,67	15
Rawon	11	100,00	0	0,00	11
Rujak Cingur	8	66,67	4	33,33	12
Sate Madura	6	85,71	1	14,29	7
Sayur Asem	11	78,57	3	21,43	14
Sayur Bayam	9	81,82	2	18,18	11
Sayur Bobor	7	87,50	1	12,50	8
Sop Ayam	13	76,47	4	23,53	17
Sop Daging	8	72,73	3	27,27	11
Soto Lamongan	9	90,00	1	10,00	10
Tahu Campur	9	100,00	0	0,00	9
Tahu Tek	5	71,43	2	28,57	7
Rata-rata	202	83,82	39	16,18	241

Berdasarkan hasil pengujian klasifikasi citra makanan dengan model *EfficientNetV2M* pada Tabel II mendapatkan hasil pengujian sistem dengan akurasi 83,82% dan kesalahan rata-rata 16,18%.

C. Uji Coba

Dalam bagian ini, uji coba dilakukan untuk mendapatkan model terbaik sehingga dapat memprediksi bahan-bahan makanan dengan cara mengganti parameter-parameter, seperti jumlah *layer*, *learning rate*, dan jumlah *frame*. Tahap awal dalam melakukkan prediksi bahan makanan yaitu membuat list bahan makanan yang tampak pada setiap gambar, kemudian membuat *dataset* proses *Transfer Learning* dengan memberi nilai 0 pada bahan makanan yang tidak terlihat dan memberi nilai 1 untuk bahan makanan yang terlihat. Kemudian memproses pickle *dataset* agar data mudah dibaca oleh sistem. Untuk memprediksi bahan-bahan makanan peneliti memanfaatkan model *EfficientNetB6* dan *EfficientNetV2M* dengan data sebanyak 1.202 data, rincian 961 data training dan 241 data testing. Dibawah ini contoh hasil prediksi bahan-bahan makanan :

```
PREDICTION:
Cabe :  0.7813321
Bawang Goreng :  0.33727598
Daun Bawang :  0.4122431
Daging Kambing :  0.92558545
==================
ACTUAL:
Tomat :  1.0
Daun Bawang :  1.0
Seledri :  1.0
Daging Kambing :  1.0
```

(a)

```
PREDICTION:
Ayam :  0.3653078
Cabe :  0.8718002
Daun Jeruk :  0.5747717
Bawang Goreng :  0.809909
==================
ACTUAL:
Tomat :  1.0
Daun Bawang :  1.0
Seledri :  1.0
Daging Kambing :  1.0
```

(b)

GAMBAR 2. Hasil Prediksi bahan-bahan Makanan (Gule Kambing)

Gambar 2 (a) adalah hasil prediksi bahan makanan pada Gule Kambing dengan model *EfficientNetV2M* yang mampu memprediksi Cabe, Bawang Goreng, Daun Bawang dan Daging Kambing. Dari hasil prediksi terdapat 2 bahan makanan yang diprediksi benar yaitu Daun Bawang dan Daging Kambing. Sedangkan gambar 2 (b) adalah hasil prediksi bahan makanan Gule Kambing dengan model *EfficientNetB6* dengan hasil prediksi Ayam, Cabe, Daun Jeruk, Bawang Goreng. Berdasarkan hasil prediksi semua bahan tidak terprediksi dengan benar.

```
PREDICTION:
Cabe :  0.55596274
Bawang Goreng :  0.6890229
Tomat :  0.8668066
Daging Kambing :  0.59799963
Daging Sapi :  0.5736612
Bawang Merah :  0.47917238
Bumbu Kecap :  0.48936975
==================
ACTUAL:
Cabe :  1.0
Bawang Goreng :  1.0
Tomat :  1.0
Bumbu Kacang :  1.0
Daging Kambing :  1.0
Bawang Merah :  1.0
Bumbu Kecap :  1.0
```

(a)

```
PREDICTION:
Daging Kambing :  0.8127581
Bumbu Kecap :  0.47282726
==================
ACTUAL:
Cabe :  1.0
Bawang Goreng :  1.0
Tomat :  1.0
Bumbu Kacang :  1.0
Daging Kambing :  1.0
Bawang Merah :  1.0
Bumbu Kecap :  1.0
```

(b)

GAMBAR 3. Hasil Prediksi bahan-bahan Makanan (Sate Madura)

Gambar 3 (a) adalah hasil prediksi bahan makanan pada Sate Madura dengan model *EfficientNetV2M* yang mampu memprediksi Cabe, Bawang Goreng, Tomat, Daging Kambing, Daging Sapi, Bawang Merah, dan Bumbu Kecap. Dari hasil prediksi terdapat 6 bahan makanan yang diprediksi benar yaitu Cabe, Bawang Goreng, Tomat, Daging Kambing, Bawang Merah, dan Bumbu Kecap. Sedangkan gambar 3 (b) adalah hasil prediksi bahan makanan Sate Madura dengan model *EfficientNetB6* dengan hasil Daging Kambing dan Bumbu Kecap. Berdasarkan hasil prediksi tidak semua bahan dapat diprediksi dengan benar, namun hanya 2 saja yang terprediksi dengan benar.

Dalam memprediksi bahan-bahan makanan dengan model terbaik dapat menampilkan hasil *confusion matrix* seperti pada Tabel III. Dari Tabel III tampak bahwa sistem dapat memprediksi bahan-bahan makanan dengan baik.

TABEL III.
HASIL UJI COBA KLASIFIKASI *MULTICLASS*

Model CNN	*Weighted avg*		
	Precision	*Recall*	*F1 Score*
EfficientNetV2M	0,65	0,65	0,64
EfficientNetB6	0,57	0,65	0,59

VI. KESIMPULAN

Berdasarkan hasil percobaan yang telah dilakukan oleh peneliti dalam mengklasifikasi citra makanan tradisional Indonesia dengan menggunakan metode DCNN model *EfficientNetV2M* dan *EfficientNetB6*. Dengan melalui beberapa tahap, pengumpulan *dataset*, *image processing*, *training*, *testing* untuk mendeteksi citra makanan dan dilanjutkan memprediksi bahan makanan. Data yang digunakan sebanyak 1.202 yang terdiri dari 961 data *training*, 241 data *testing*. Sistem dapat mengklasifikasi citra makanan sehingga dapat memprediksi makanan tradisional Indonesia dengan akurasi 83,82% dan kesalahan 16,18% sedangkan model *EfficientNetB6* tingkat akurasinya 80,08% dan kesalahan 19,92% dengan Sedangkan dalam memprediksi bahan-bahan makanan dengan model *EfficientNetB6* mendapatkan nilai *F1Score* 59%, *Recall* 65%, *Precission* 57%, dan untuk model *EfficientNetV2M* mendapatkan nilai *F1Score* 64%, *Recall* 65%, *Precission* 65%, sehingga dapat disimpulkan bahwa model *EfficientNetV2M* lebih baik dari model *EfficientNetB6* dalam mengklasifikasikan citra makanan dan memprediksi bahan-bahan makanan.

PERAN PENULIS

Citra Mahaputri: Konseptualisasi, metodologi, perangkat lunak, validasi, investigasi, sumber daya, kurasi data, penyusunan draft asli, visualisasi;
Yosi Kristian: Konseptualisasi, metodologi, perangkat lunak, validasi, analisis formal, investigasi, penyusunan draft asli, peninjauan dan penyuntingan, visualisasi, pengawasan, administrasi proyek;
Endang Setyati: Validasi, analisi formal, investigasi,

peninjauan dan penyuntingan, pengawasan, administrasi proyek;

DAFTAR PUSTAKA

[1] I. Pamelia, "Perilaku Konsumsi Makanan Cepat Saji Pada Remaja Dan Dampaknya Bagi Kesehatan," *Ikesma*, vol. 14, no. 2, p. 144, 2018, doi: 10.19184/ikesma.v14i2.10459.

[2] Y. Mursono, "Prospek Pengembangan Makanan Fungsional," *J. Teknol. Pangan dan Gizi*, vol. 7, no. 1, pp. 19–27, 2007, [Online]. Available: http://elearning.unsri.ac.id/pluginfile.php/635/mod_forum/attachment/23137/ipi113801.pdf.

[3] A. Baequny, A. S. Harnany, and E. Rumimper, "Pengaruh Pola Makan Tinggi Kalori terhadap Peningkatan Kadar Gula Darah pada Penderita Diabetes Mellitus Tipe 2," *J. Ris. Kesehat.*, vol. 4, no. 1, pp. 687–692, 2015, [Online]. Available: http://ejournal.poltekkes-smg.ac.id/ojs/index.php/jrk/article/view/347.

[4] G. A. Pamungkas, R. R. Isnanto, and K. T. Martono, "Pembuatan Aplikasi Panduan Gizi Seimbang Berbasis Android Dengan Menggunakan Metode Backward Chaining," *J. Teknol. dan Sist. Komput.*, vol. 4, no. 2, p. 369, 2016, doi: 10.14710/jtsiskom.4.2.2016.369-379.

[5] Z. Shen, "ScienceDirect Machine Learning Based Approach on Food Recognition Approach on Food Recognition Machine Based Approach on and Nutrition Estimation Machine Learning Learning Based Approach on Food Food Recognition Recognition Machine Learning Based Approach," vol. 100, pp. 1–6, 2019.

[6] J. Teng, D. Zhang, D. J. Lee, and Y. Chou, "Recognition of Chinese food using convolutional neural network," *Multimed. Tools Appl.*, vol. 78, no. 9, pp. 11155–11172, 2019, doi: 10.1007/s11042-018-6695-9.

[7] D. Kraft and G. Bieber, "Vietnamese Food Recognition System Using Convolutional Neural Networks Based Features," *ACM Int. Conf. Proceeding Ser.*, vol. 3, pp. 423–428, 2020, doi: 10.1145/3389189.3397993.

[8] I. P. A. E. Darma Udayana, M. Sudarma, and P. G. Surya Cipta Nugraha, "Implementation of Convolutional Neural Networks to Recognize Images of Common Indonesian Food," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 846, no. 1, 2020, doi: 10.1088/1757-899X/846/1/012023.

[9] Sandhopi, Lukman Zaman P.C.S.W, and Yosi Kristian, "Identifikasi Motif Jepara pada Ukiran dengan Memanfaatkan Convolutional Neural Network," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 9, no. 4, pp. 403–413, 2020, doi: 10.22146/jnteti.v9i4.541.

[10] Y. Kristian, I. K. E. Purnama, E. H. Sutanto, L. Zaman, E. I. Setiawan, and M. H. Purnomo, "Klasifikasi Nyeri pada Video Ekspresi Wajah Bayi Menggunakan DCNN Autoencoder dan LSTM," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 7, no. 3, pp. 308–316, 2018, doi: 10.22146/jnteti.v7i3.440.

[11] S. J. Park, A. Palvanov, C. H. Lee, N. Jeong, Y. I. Cho, and H. J. Lee, "The development of food image detection and recognition model of Korean food for mobile dietary management," *Nutr. Res. Pract.*, vol. 13, no. 6, pp. 521–528, 2019, doi: 10.4162/nrp.2019.13.6.521.

[12] J. Sun, K. Radecka, and Z. Zilic, "Exploring better food detection via transfer learning," *Proc. 16th Int. Conf. Mach. Vis. Appl. MVA 2019*, pp. 1–6, 2019, doi: 10.23919/MVA.2019.8757886.

[13] N. Martinel, G. L. Foresti, and C. Micheloni, "Wide-slice residual networks for food recognition," *Proc. - 2018 IEEE Winter Conf. Appl. Comput. Vision, WACV 2018*, vol. 2018-Janua, pp. 567–576, 2018, doi: 10.1109/WACV.2018.00068.

[14] S. R. Suartika E. P, I Wayan, Wijaya Arya Yudhi, "Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) Pada Caltech 101," *J. Tek. ITS*, vol. 5, no. 1, p. 76, 2016, [Online]. Available: http://repository.its.ac.id/48842/.

[15] A. Peryanto, A. Yudhana, and R. Umar, "Klasifikasi Citra Menggunakan Convolutional Neural Network dan K Fold Cross Validation," *J. Appl. Informatics Comput.*, vol. 4, no. 1, pp. 45–51, 2020, doi: 10.30871/jaic.v4i1.2017.

[16] C. Y. Lee, P. W. Gallagher, and Z. Tu, "Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree," *Proc. 19th Int. Conf. Artif. Intell. Stat. AISTATS 2016*, pp. 464–472, 2016.

[17] M. A. Abu, N. H. Indra, A. H. A. Rahman, N. A. Sapiee, and I. Ahmad, "A study on image classification based on deep learning and tensorflow," *Int. J. Eng. Res. Technol.*, vol. 12, no. 4, pp. 563–569, 2019.

[18] K. Seetala, W. Birdsong, and Y. B. Reddy, "Image classification using tensorflow," *Adv. Intell. Syst. Comput.*, vol. 800 Part F, no. Itng, pp. 485–488, 2019, doi: 10.1007/978-3-030-14070-0_67.

[19] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, *A Survey on Deep Transfer Learning Chuanqi*, vol. 11141, no. November. Springer International Publishing, 2018.

[20] Y. Wu, X. Qin, Y. Pan, and C. Yuan, "Convolution neural network based transfer learning for classification of flowers," *2018 IEEE 3rd Int. Conf. Signal Image Process. ICSIP 2018*, pp. 562–566, 2019, doi: 10.1109/SIPROCESS.2018.8600536.

[21] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.

[22] A. Rohim, Y. A. Sari, and Tibyani, "Convolution neural network (cnn) untuk pengklasifikasian citra makanan tradisional," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 7, pp. 7038–7042, 2019, [Online]. Available: http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/5851/2789.

Pengenalan Ekspresi Wajah dengan CNN dan Wavelet

Erwin Sentosa [1], Hendrawan Armanto [1], C. Pickerling [1], Lukman Zaman PCSW [1]
[1]Departemen Informatika, Fakultas Sains dan Teknologi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Indonesia

Corresponding author: Erwin Sentosa (e-mail: erwinsentosa1@gmail.com).

ABSTRACT With the development of technology in modern times, it is hoped that computers will also be able to recognize human facial expressions with advances in machine learning. Machine learning has become a part of everyday life for many people worldwide. The discovery and implementation of machine learning allow computers to learn and predict possible patterns and can be used to help humans perform daily activities. One of them is Convolutional Neural Network. In this study, wavelet transform will be used to help improve the accuracy of the convolutional neural network and accelerate the increase in accuracy. Wavelets help compress images so that they are easier to process. The image generated by the wavelet is divided into 4 different frequencies. Each image generated by the wavelet is tested into a convolutional neural network. Based on the experiments' results, the best accuracy was obtained from the KDEF dataset using Low-Low (LL) frequency wavelet images with an accuracy of 79%. While the results of trials using self-made datasets obtained the best accuracy using Low-Low (LL) frequency wavelets with an accuracy of 36.925%.

KEYWORDS Convolutional Neural Network, Facial Expression, Machine Learning, Wavelet

ABSTRAK Dengan berkembangnya teknologi di jaman modern ini diharapkan komputer juga mampu mengenali ekspresi wajah manusia. Hal itu dapat terwujud dengan kemajuan machine learning. Machine learning telah menjadi bagian dari kehidupan sehari-hari bagi banyak orang di seluruh dunia. Penemuan dan implementasi machine learning memungkinkan komputer mempelajari dan memprediksi pola yang mungkin terjadi dan dapat digunakan untuk membantu manusia dalam aktivitas sehari-hari. Salah satunya yaitu Convolutional Neural Network. Pada penelitian ini akan digunakan wavelet transform untuk membantu meningkatkan akurasi dari convolutional neural network dan mempercepat peningkatan akurasi. Wavelet berguna untuk melakukan compressing pada gambar sehingga lebih mudah untuk diolah. Gambar yang dihasilkan oleh wavelet terbagi menjadi 4 frekuensi yang berbeda-beda. Setiap gambar yang dihasilkan oleh wavelet diuji cobakan kedalam convolutional neural network. Berdasarkan hasil uji coba yang dilakukan, akurasi terbaik didapatkan dari dataset KDEF dengan menggunakan gambar wavelet berfrekuensi Low-Low (LL) dengan akurasi yang didapatkan sebesar 79%. Sedangkan hasil uji coba menggunakan dataset buatan sendiri didapatkan akurasi terbaik dengan menggunakan wavelet berfrekuensi Low-Low (LL) dengan akurasi yang didapatkan sebesar 36,925%.

KATA KUNCI *Convolutional Neural Network,* Ekspresi Wajah *, Machine Learning, Wavelet* .

I. PENDAHULUAN

Wajah merupakan bagian dari tubuh manusia yang menjadi fokus di dalam interaksi sosial. Wajah juga merupakan salah satu bagian unik dari tubuh manusia yang mempunyai karakteristik yang berbeda. Wajah memiliki peranan penting dengan menunjukkan emosi/ekspresi. Wajah manusia menyediakan banyak informasi, banyak hal menarik yang bisa diperhatikan, dan dipelajari secara intensif.

Beberapa penelitian yang menggali informasi dari wajah manusia adalah pengenalan wajah dan pengenalan isyarat wajah. Ketika manusia berinteraksi satu sama lain, mereka menggunakan berbagai macam isyarat dari wajah untuk menyampaikan informasi. Isyarat wajah yang terbentuk bisa menyampaikan ekspresi wajah tertentu.

Dengan berkembangnya teknologi di jaman modern ini diharapkan komputer juga mampu mengenali ekspresi wajah

manusia. Ini dapat diwujudkan melalui kemajuan machine learning. Machine learning telah menjadi bagian dari kehidupan sehari-hari bagi banyak orang di seluruh dunia. Penemuan dan implementasi machine learning memungkinkan komputer mempelajari dan memprediksi pola yang mungkin terjadi dan dapat digunakan untuk membantu manusia dalam aktivitas sehari-hari. Salah satunya yaitu Convolutional Neural Network.

CNN adalah jenis neural network yang biasa digunakan untuk pengolahan data gambar. CNN dapat digunakan untuk mendeteksi dan mengidentifikasi objek pada citra. Secara umum CNN adalah varian dari Neural Network (NN) yang dibangun untuk memproses data dalam jumlah besar, sehingga prosesnya lebih efisien daripada menggunakan NN. CNN terdiri dari neuron yang memiliki weight, bias dan activation function. CNN dapat mempermudah pengklasifikasian ekspresi wajah yang sebelumnya cukup susah, dikarenakan pengambilan gambar dapat dari berbagai macam sudut dan perbedaan wajah dari setiap orang yang ada yang menyebabkan sulitnya untuk menemukan persamaan ekspresi dari setiap wajah orang.

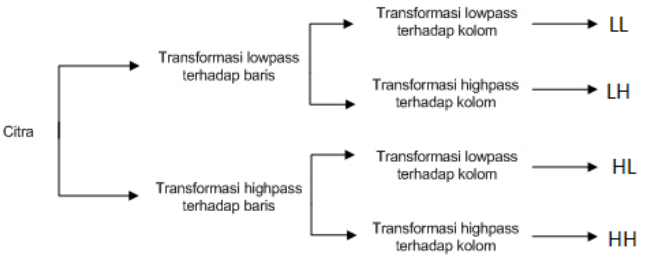
Tujuan utama dari penelitian ini adalah untuk membuat membuktikan penggunaan wavelet untuk preprocessing gambar dan CNN yang dapat mengklasifikasikan ekspresi wajah manusia berdasarkan gambar/foto wajah manusia yang diambil dari berbagai macam sudut sebagai input dan output berupa teks [1].

II. TEORI PENUNJANG

1. *Wavelet*

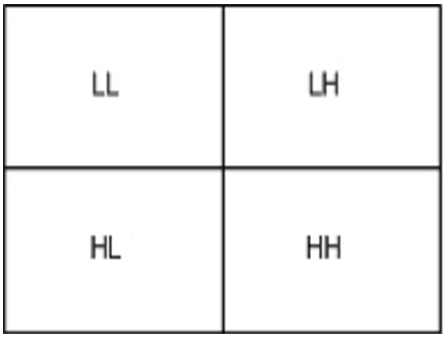
Transformasi Wavelet Diskrit adalah proses fungsi wavelet yang memilih subset dari skala dan titik tertentu dalam proses komputasi. Dalam Transformasi ini, sinyal citra dapat dianalisis dengan melewatkannya melalui proses filtering. Proses filtering tersebut terdiri dari low-pass dan high-pass filter pada setiap langkah dekomposisi. Filtering dilakukan pada baris dan kolom.

Salah satu jenis dari transformasi wavelet diskrit yaitu haar wavelet. Wavelet Haar merupakan wavelet yang paling tua dan sederhana. Citra asli akan didekomposisi menjadi 4 gambar dengan frekuensi yang berbeda-beda. Gambar asli akan difilter bedasarkan baris dan kolom. Filter yang digunakan ada 2 macam yaitu low-pass dan high-pass. Berikut ilustrasi filtering dapat dilihat pada gambar di bawah ini.



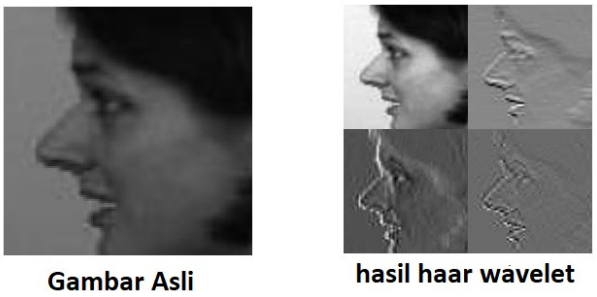
GAMBAR 1. Illustrasi Filtering Wavelet

Pada gambar diatas dapat dilihat terdapat 4 macam frekuensi yang dihasilkan dari proses filtering. Frekuensi yang akan dihasilkan yaitu Low-Low(LL), Low-High(LH), High-Low(HL), dan High-High(HH). Low-Low frekuensi dihasilkan dari hasil low-pass filtering terhadapa baris dan kolom, Low-High frekuensi dihasilkan dari low-pass filtering terhadap baris dan high-pass filtering terhadap kolom, High-Low frekuensi dihasilkan dari high-pass filtering terhadap baris dan low-pass filtering terhadap kolom, dan High-High frekuensi dihasilkan dari high-pass filtering terhadap baris dan kolom. Berikut gambar ilustrasi output yang dihasilkan dapat dilihat pada gambar di bawah ini [2].



GAMBAR 2. Illustrasi Output Wavelet

Setelah citra didekomposisi dengan melakukan langkah-langkah diatas maka akan menghasilkan citra baru dengan 4 macam frekuensi yang berbeda-beda. Output dari proses tersebut akan terlihat gambar terbagi menjadi 4 bagian dengan ukuran ¼ dari gambar asli seperti pada gambar 2.2. Berikut contoh hasil proses dekomposisi haar wavelet dapat dilihat pada gambar 3 di bawah ini

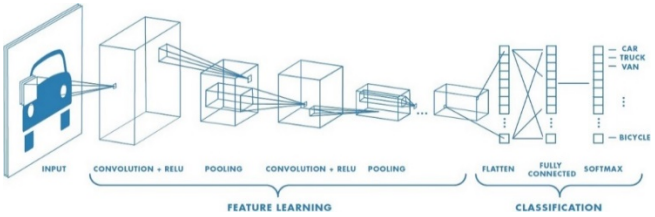


GAMBAR 3. Hasil Wavelet

2. *Convolutional Neural Network*

CNN adalah pengembangan Multi Layer Perceptron (MLP) untuk memproses data dua dimensi. CNN diklasifikasikan sebagai jenis Deep Neural Network karena kedalaman jaringannya yang besar dan banyak digunakan untuk data citra. Dalam permasalahan klasifikasi citra, MLP tidak cocok karena tidak menyimpan informasi spasial dan mengasumsikan bahwa setiap pixel adalah fitur terpisah sehingga memberikan hasil yang buruk.

Asal mula penelitian yang menjadi dasar penemuan ini dilakukan oleh Hubel dan Wiesel. Mereka melakukan penelitian visual korteks pada penglihatan kucing. Penelitian ini sangat berguna dalam sistem pemrosesan visual yang ada. Sejak penelitian tersebut, banyak penelitian yang terinspirasi oleh mekanisme yang disajikan dan menghasilkan model-model baru seperti Neocognitron, HMAX, dan LeNet-5. Arsitektur CNN dibagi menjadi 2 bagian utama, Feature Learning/Extraction Layer dan Classification Layer, seperti yang ditunjukan pada Gambar 4. [3]



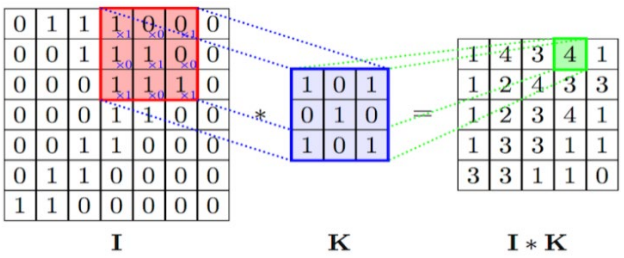
GAMBAR 4. Ilustrasi Filtering Wavelet

Cara kerja CNN mirip dengan MLP, namun pada CNN setiap neuron direpresentasikan dalam dua dimensi, berbeda dengan MLP dimana setiap neuron hanya satu dimensi. Pada CNN, data yang disebarkan melalui jaringan selalu data dua dimensi, sehingga operasi linier dan bobot parameter CNN berbeda dengan MLP. Operasi Linier CNN menggunakan konvolusi, sedangkan bobot tidak lagi hanya satu dimensi, tetapi dalam bentuk empat dimensi yang mewakili kumpulan filter konvolusi. Karena sifat dari proses konvolusi, maka CNN hanya dapat digunakan untuk data yang memiliki struktur dua dimensi, seperti gambar.

1. *Convolutional Layer*

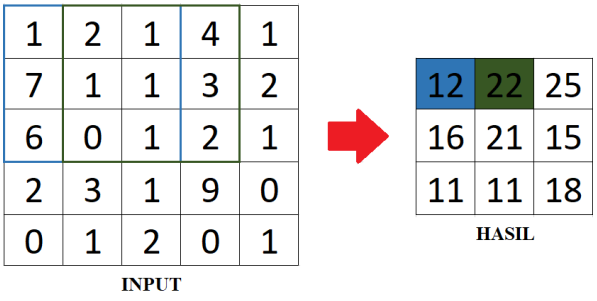
Convolutional Layer adalah salah satu jenis layer yang terdapat di dalam CNN dan dapat dikatakan sebagai layer inti. Convolutional Layer melakukan operasi konvolusi yang dimana operasi konvolusi ini merupakan sebuah kunci utama atau operasi utama dari CNN. Layer ini memiliki tujuan untuk melakukan ekstraksi untuk mendapatkan fitur-fitur penting

Dalam Convolutional Layer sendiri, terdapat sebuah proses utama yang biasa disebut dengan konvolusi, dimana konvolusi adalah istilah matematika yang berarti menerapkan fungsi berulang kali ke output fungsi lain. Berikut merupakan sebuah gambar yang menunjukkan sebuah contoh dari ilustrasi Convolutional Layer [4].



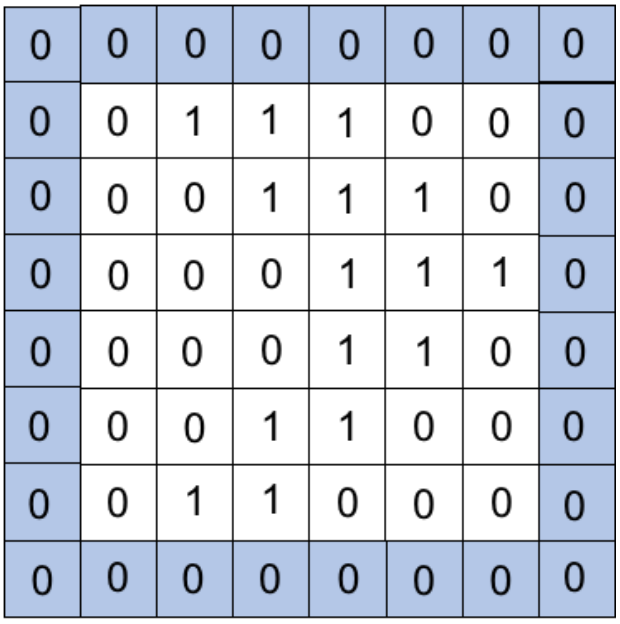
GAMBAR 5. Contoh Convolutional Layer

Pada convolutional layer terdapat beberapa hyperparameter antara lain stride, padding, dan fungsi aktivasi. Stride adalah parameter yang menentukan seberapa jauh jumlah pergeseran filter. Nilai 1, akan menggeser filter sebanyak 1 pixels. Nilai dari stride berlaku untuk pergeseran filter secara horisontal maupun secara vertical. Semakin kecil stride, semakin detail informasi yang didapatkan dari input, tetapi akan membutuhkan lebih banyak perhitungan dibandingkan dengan stride besar. Lihat gambar 6 untuk illustrasi stride 1.



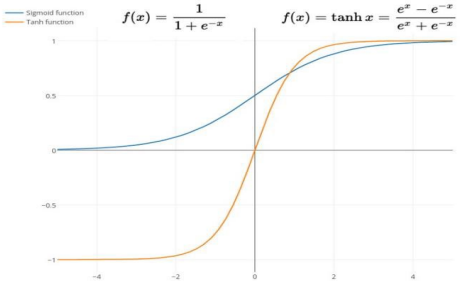
GAMBAR 6. Ilustrasi Stride 1

Padding adalah parameter yang menentukan jumlah pixel untuk ditambahkan ke setiap sisi input. Padding yang biasa digunakan yaitu zero padding. Zero padding akan menambahkan angka 0 pada setiap sisi input. Contoh zero padding dapat dilihat pada gambar 7 dibawah ini



GAMBAR 7. Zero Padding

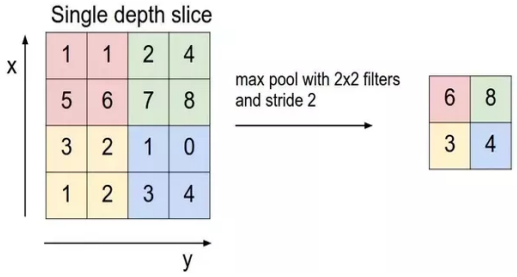
Fungsi aktivasi menentukan apakah sebuah neuron harus aktif atau tidak berdasarkan bobot total input. Secara umum, ada 2 jenis fungsi aktivasi, yaitu fungsi aktivasi linear dan non linear. Contoh dari fungsi aktivasi non linear antara lain Rectified Liniear Unit (ReLU), Softmax, dan Sigmoid, Tanh (lihat Gambar 8).



GAMBAR 8. Zero Padding

2. *Pooling Layer*

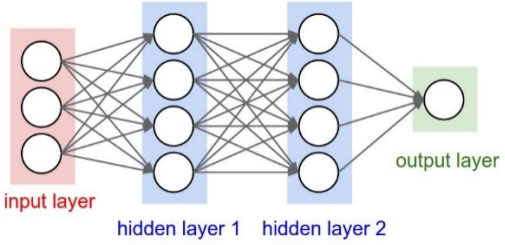
Pooling layer adalah proses untuk memperkecil ukuran citra. Dalam pengolahan citra, pooling juga bertujuan untuk meningkatkan invariansi karakteristik dari fitur. Max pooling adalah metode pooling yang digunakan di sebagain besar CNN. Max pooling membagi output dari convolution layer menjadi beberapa grid kecil kemudian mengambil nilai maksimum dari setiap grid untuk membentuk matriks citra yang lebih kecil. Contoh ilustrasi max pooling layer dapat dilihat pada gambar 9.



GAMBAR 9. Max Pooling

3. *Fully Connected Layer*

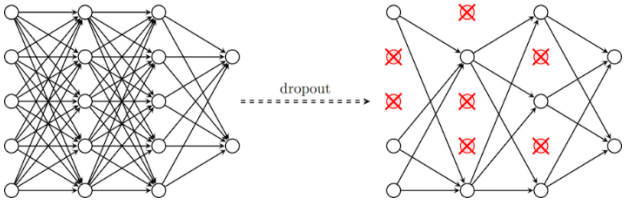
Fully connected layer adalah layer yang umum digunakan dalam MLP yang tujuannya untuk mentransformasi dimensi data agar data dapat diklasifikasikan secara linear. Pada CNN fully connected layer biasanya dilakukan setelah gambar sudah diproses pada convolutional layer dan pooling layer. Setiap neuron pada convolutional layer pertama-tama harus dikonversi menjadi data satu dimensi sebelum dapat dimasukkan ke dalam sebuah fully connected layer. Karena data kehilangan informasi spasialnya dan tidak dapat dibalik, maka fully connected layer hanya dapat diimplementasikan pada bagian akhir. Gambar 10 adaah Illustrasi fully connected layer.



GAMBAR 10. Fully Connected Layer

4. *Dropout*

Dropout adalah proses untuk mencegah overfitting dan mempercepat proses pembelajaran. Dropout mengacu pada penghapusan neuron yang tersembunyi ataupun terlihat dalam jaringan. Melakukan penghapusan neuron, berarti menghilangkan sementara dari jaringan yang ada. Neuron yang akan dihapus dipilih secara random dengan probabilitas tertentu. Setiap neuron diberi p-probabilitas yang bernilai mulai dari 0 hingga 1.0. Untuk lebih jelasnya, ilustrasi mengenai proses dropout dapat di lihat pada gambar di bawah ini [5].

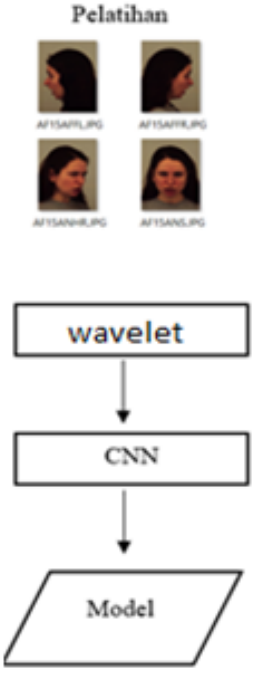


GAMBAR 11. Dropout

III. Arsitektur [6]–[9]

A. Arsitektur Sistem

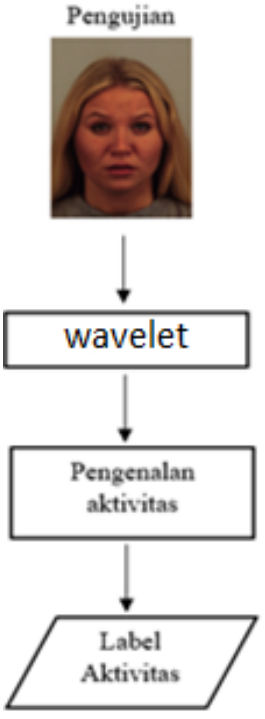
Struktur sistem yang akan dibuat secara garis besar akan dibagi menjadi 2 bagian yaitu bagian pelatihan dan bagian uji coba. Pada tahap pertama sistem akan melakukan proses pelatihan terlebih dahulu. Hal ini bertujuan agar nantinya dapat mengklasifikasikan ekspresi manusia secara tepat pada saat dilakukan prosses uji coba. Agar lebih jelas struktur tahap pelatihan dapat dilihat pada gambar 12 di bawah ini.



GAMBAR 12. Struktur Proses Pelatihan

Proses pelatihan dimulai dengan memasukan dataset. Dataset yang digunakan untuk training berupa kumpulan gambar wajah manusia yang memiliki ekspresi berbeda-beda. Dataset tersebut akan terlebih dahulu diproses menggunakan wavelet agar nantinya dapat diproses lebih cepat didalam CNN dan diharapkan dapat menaikkan akurasi. Gambar – gambar tersebut akan dimasukkan kedalam CNN untuk melakukan proses klasifikasi.

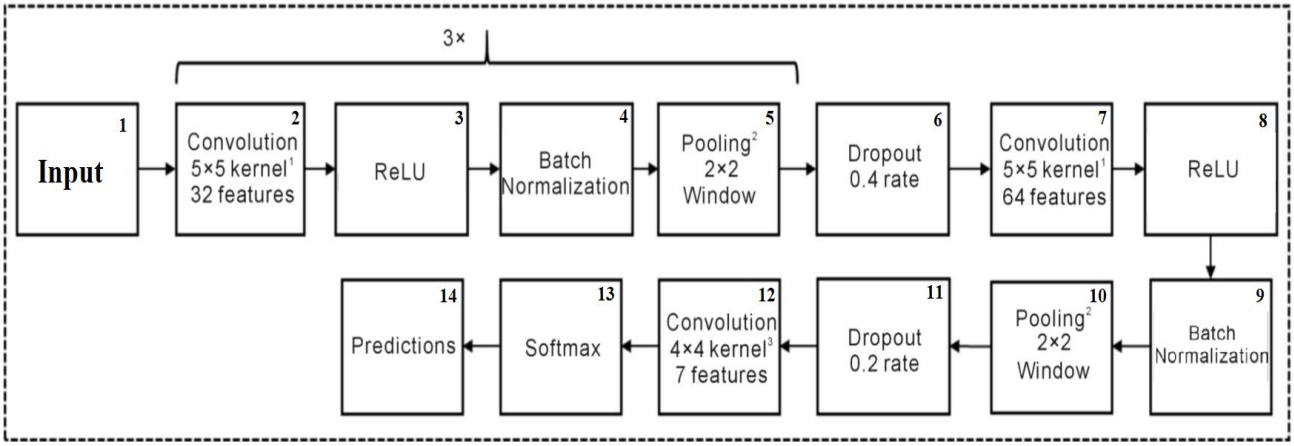
Model bobot filter yang didapatkan dari hasil proses pelatihan akan disimpan. Bobot ini kemudian akan dipakai ketika melakukan proses uji coba. Untuk struktur uji coba dapat dilihat pada gambar 13



S

GAMBAR 13. Struktur Proses Uji Coba

Dapat kita lihat pada gambar diatas, setelah gambar dimasukkan kedalam CNN pada proses uji coba, program akan mendapatkan output berupa label dari ekspresi tersebut. Label tersebut berupa ekspresi dari wajah manusia yang diinputkan kedalam program pada saat proses uji coba. Tahap ini merupakan tujuan utama dari program ini yaitu untuk mengenali ekspresi wajah manusia.



GAMBAR 14. Arsitektur CNN

B. Arsitektur CNN

Arsitektur CNN merupakan struktur susunan layer yang akan digunakan untuk mengklasifikan data dua dimensi. Arsitektur CNN yang dibuat akan berupa susunan layer-layer pada CNN. Dalam pembuatan program ini arsitektur akan dibuat sedemikian rupa agar dapat mengklasifikasikan ekspresi wajah manusia dengan baik.

Arsitektur CNN yang akan digunakan terdiri dari convolutional layer, normalization, ReLU, pooling layer, dropout, dan softmax. Layer-layer tersebut disusun sedemikian rupa sehingga dapat mengklasifikan ekspresi wajah dengan tingkat akurasi yang tinggi. Untuk lebih jelasnya arsitektur CNN dapat dilihat pada gambar 14 di atas.

Dapat kita lihat pada arsitektur di gambar 14, terdapat 14 langkah dalam proses pengklasifikasian yang akan dijalankan. Setiap langkah yang dilakukan terdapat fungsi atau kegunaannya masing-masing. Fungsi dari setiap langkahnya akan membantu dalam proses klasifikasi yang akan dilakukan.

Langkah pertama yang dilakukan adalah menerima inputan. Inputan tersebut kemudian akan diproses ke dalam convolution layer dan data yang dihasilkan oleh convolutional layer akan diproses ke dalam ReLU. Setelah itu data yang di hasilkan dari ReLU akan diproses kedalam batch normalization dan pooling layer. Seperti yang dapat kita lihat pada gambar arsitektur CNN di atas, langkah nomor 2 hingga nomor 5 akan diulang sebanyak 3 kali.

Data-data yang dihasilkan dari proses diatas akan dimasukan ke dalam layer dropout. kemudian data yang didapat dari proses dropout akan dimasukkan ke dalam convolution layer, ReLU, dan batch normalization. Setelah

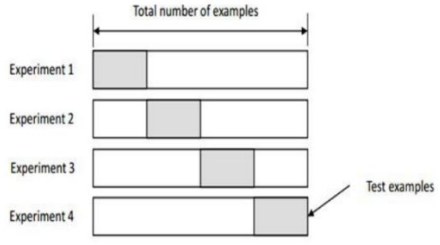
itu data yang di hasilkan akan dimasukkan kedalam dropout layer lagi. Sama seperti sebelumnya data yang dihasilkan setelah proses dropout akan dimasukkan kembali kedalam convolutional layer, akan tetapi dengan nilai parameter convolutional layer yang berbeda dengan sebelumnya. Hasil dari convolutional layer tersebut kemudian akan dimasukkan kedalam softmax dan setelah itu kita dapat mendapatkan hasil yang didapat pada tahap prediction. Untuk lebih jelasnya langkah-langkah arsitektur CNN pada gambar 14 dapat dilihat pada tabel di bawah ini.

TABLE I
ARSITEKTUR CNN

LAYER	FILTER SIZE	OUTPUT DEPTH	ACTIVATION FUNCTION
Input	-	-	-
Convolution Layer	5x5	32	ReLU
Batch Normalization	-	-	-
Max Pooling	2x2	-	-
Convolutional Layer	5x5	32	ReLU
Batch Normalization	-	-	-
Max Pooling	2x2	-	-
Convolutional Layer	5x5	32	ReLU
Batch Normalization	-	-	-
Max Pooling	2x2	-	-
Dropout	-	-	-
Convolutional Layer	5x5	64	ReLU
Batch Normalization	-	-	-
Max Pooling	2x2	-	-
Dropout	-	-	-
Convolutional Layer	4x4	7	Softmax

IV. *UJI COBA*

Uji coba dilakukan dengan menggunakan K-fold cross validation. K-fold cross validation merupakan metode uji coba dengan membagi data menjadi K bagian, kemudian akan diulang sebanyak K dengan mengganti bagian yang manjadi data testing. Untuk lebih jelasnya dapat dilihat pada gambar di bawah ini [10].



GAMBAR 15. K-fold cross validation

Gambar di atas merupakan contoh illustrasi k-fold cross validation dengan nilai K=4. Data yang dimiliki akan dibagi menjadi 4 bagian. ¼ bagian akan menjadi data testing dan ¾ bagian akan menjadi data training. Kemudian untuk setiap iterasinya data testing akan berubah-ubah. Karena nilai K=4 maka jumlah uji coba yang dilakukan sebanyak 4 kali.

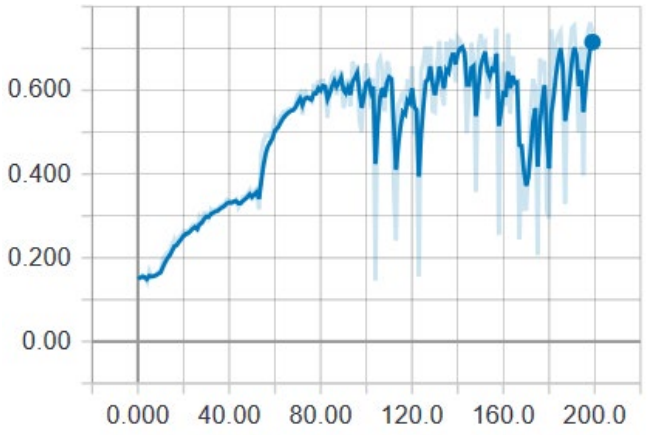
Dataset yang digunakan pada proses uji coba berupa foto wajah manusia dengan 7 macam ekspresi. Ketujuh macam ekspresi tersebut yaitu netral, senang, marah, sedih, takut, jijik, dan kaget. untuk lebih jelasnya gambar ekspresi wajah dapat dilihat pada gambar di bawah ini

GAMBAR 16. Ekspresi Dataset

Selain memiliki 7 ekspresi, dataset yang digunakan juga diambil dari 5 sudut pengambilan foto yang berbeda. Sudut pengambilan foto tersebut yaitu kiri, serong kiri, depan, serong kanan, dan kanan, untuk lebih jelasnya dapat dilihat pada gambar di bawah ini.

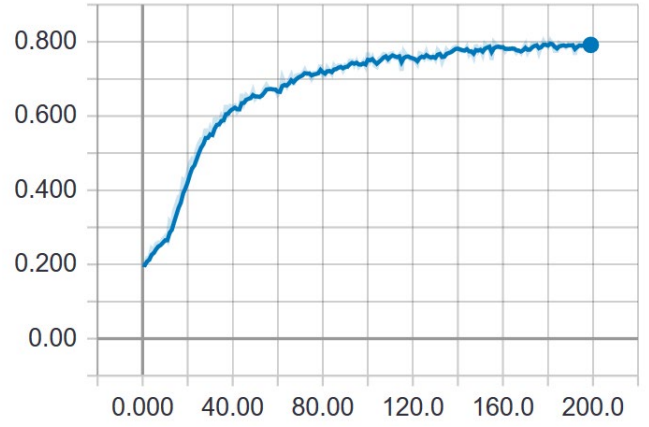
GAMBAR 17. Sudut Pengambilan Foto Dataset

Data yang digunakan dalam proses uji coba yaitu data gambar gray scale dengan menggunakan dataset KDEF. Uji coba ini dilakukan sebanyak 200 ecpoch dan dilakukan sebanyak 4 kali menggunakan metode k-fold cross validation dengan nilai K=4. Berikut ini merupakan gambar dari grafik akurasi paling baik yang dihasilkan pada uji coba ini.



GAMBAR 18. Grafik Akurasi Gray Scale

Seperti yang dapat dilihat pada gambar di atas, grafik yang tampilkan menunjukkan bahwa akurasi yang didapatkan dari data testing selama proses training berlangsung memiliki nilai akurasi tertinggi 73%. Gambar grafik di bawah merupakan salah contoh akurasi yang didapatkan dari salah satu proses uji coba yang dilakukan terhadap data asli. Kemudian dari hasil uji coba lainnya didapatkan nilai akurasi tertinggi mencapai 71%, 68,8% dan 72%. Sehingga dari hasil coba tersebut didapatkan akurasi rata-rata sebesar 71,2%. Setelah itu akan diuji cobakan gambar Low-Low frekuensi yang dihasilkan dari proses wavelet. Grafik akurasi yang dihasilkan dapat dilihat pada gambar di bawah ini



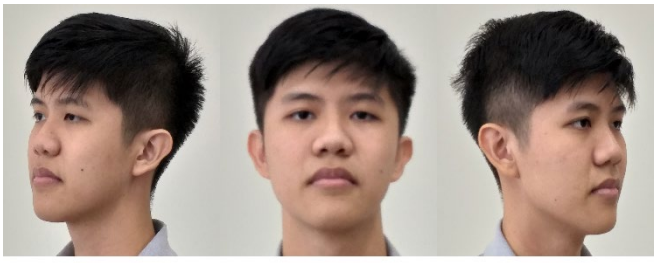
GAMBAR 19. Grafik Akurasi Low-Low

Seperti yang dapat dilihat pada gambar di atas, grafik akurasi pada data testing Low-Low Frekuensi menujukkan bahwa dari awal proses training, grafik menunjukkan kenaikan yang cukup baik hingga mencapai akurasi sekitar 75%. Setelah itu, akurasi mulai stabil pada 75% hingga mencapai titik tertingginya yaitu 80%. Hasil uji coba lainnya mendapatkan akurasi sebesar 79,2%, 77%, dan 78% sehingga hasil rata-rata dari ke empat uji coba tersebut sebsar 78,5%.

Selain menggunakan dataset yang telah dijelaskan sebelumnya, juga di lakukan proses uji coba dengan menggunakan dataset yang dibuat sendiri. Dataset buatan sendiri tersebut memiliki 7 ekspresi yaitu netral, senang, marah, sedih, jijik, takut, dan kaget. Untuk lebih jelasnya ekspresi wajah tersebut dapat dilihat pada gambar di bawah ini.

GAMBAR 20. Ekspresi Wajah Dataset Buatan Sendiri

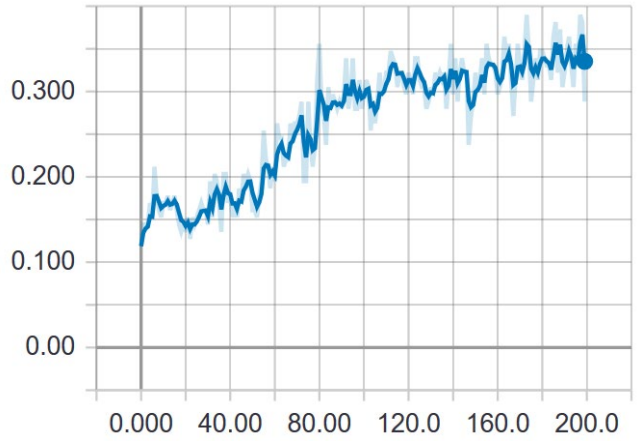
Selain memiliki 7 ekspresi, dataset yang digunakan juga diambil dari 3 sudut pengambilan foto yang berbeda. Sudut pengambilan foto tersebut yaitu serong kiri, depan, dan serong kanan. Untuk lebih jelasnya dapat dilihat pada gambar di bawah ini



GAMBAR 21. Sudut Pengambilan Foto Dataset Buatan Sendiri

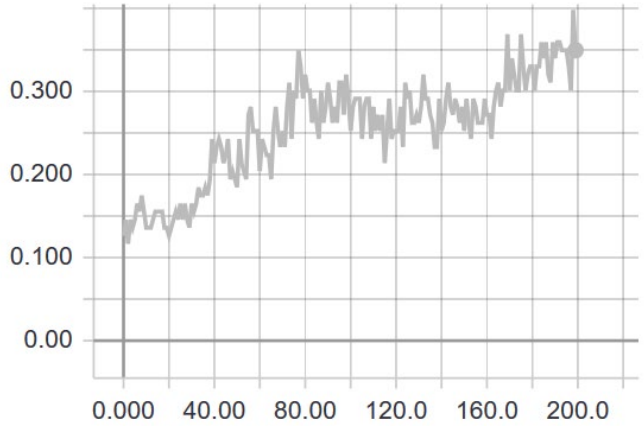
Sama dengan dataset sebelumnya dilakukan uji coba dengan menggunakan gambar asli yang di gray scale. Uji coba ini dilakukan sebanyak 200 epoch dengan menggunakan k-fold cross validation dengan nilai K=4. Berikut ini merupakan grafik yang didapatkan dari hasil uji coba tersebut.

Seperti yang dapat dilihat pada gambar di atas, grafik yang tampilkan menunjukkan bahwa akurasi yang didapatkan dari data testing selama proses training berlangsung memiliki nilai akurasi tertinggi 38,8%. Setelah itu dari tiga proses uji coba lainnya didapatkan akurasi tertinggi sebsar 35%, 32,2%, dan 30% sehingga didapatkan hasil rata-rata dari semua uji coba tersebut sebsar 34,025%. Setelah itu akan diuji cobakan gambar Low-Low frekuensi yang dihasilkan dari proses wavelet. Grafik akurasi yang dihasilkan dapat dilihat pada gambar di bawah ini.



GAMBAR 22. Grafik Akurasi Gambar Asli Dataset Buatan Sendiri

Seperti yang dapat dilihat pada gambar di atas, grafik akurasi pada data testing Low-Low Frekuensi menujukkan bahwa dari awal proses training, grafik menunjukkan kenaikan tetapi tidak terlalu stabil. Dari grafik di atas didapatkan akurasi maximal sebesar 39.8%. Kemudian dari 3 uji coba lainnya didapatkan akurasi tertinggi sebesar 35,9%, 38,1%, dan 33,9% sehingga mendapatkan hasil rata-rata akurasi sebesar 36,925%



GAMBAR 23. Grafik Akurasi Low-Low Dataset Buatan Sendiri

V. KESIMPULAN

Pada bagian ini akan dijelaskan mengenai kesimpulan apa saja yang didapatkan dalam penelitian ini. Kesimpulan-kesimpulan yang didapatkan akan dijelaskan dalam beberapa poin di bawah ini.

- Dengan menggunakan haar Wavelet transform hasil akurasi yang didapatkan lebih tinggi dibandingkan dengan tanpa menggunakan haar wavelet transform.
- Akurasi terbaik yang didapatkan yaitu sebesar 79% dimana data yang digunakan merupakan data hasil dari haar wavelet transform yang berfrekuensi Low-Low.
- Dataset buatan sendiri memiliki akurasi rendah dibandingkan dataset KDEF, dikarenakan ketidak stabilan pengambilan gambar. Mayoritas gambar yang diperoleh tidak konsisten. Sedangkan dataset KDEF, memiliki data yang konsisten untuk seluruh orang.

PERAN PENULIS

Setiap penulis memiliki kontribusi yang sama dalam Analisis Formal, Investigasi, Administrasi Proyek, Sumber Daya, Perangkat Lunak, Validasi, Visualisasi, Penulisan Penyusunan Draf Asli, Penulisan Review & Penyuntingan.

COPYRIGHT

DAFTAR PUSTAKA

[1] T. Williams, R. Li, and others, "An ensemble of convolutional neural networks using wavelets for image classification," *J. Softw. Eng. Appl.*, vol. 11, no. 02, p. 69, 2018.

[2] L. Novamizanti and A. Kurnia, "Analisis Perbandingan Kompresi Haar Wavelet Transform dengan Embedded Zerotree Wavelet pada Citra," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. \& Tek. Elektron.*, vol. 3, no. 2, p. 161, 2015.

[3] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," *CoRR*, vol. abs/1511.0, 2015, [Online]. Available: http://arxiv.org/abs/1511.08458.

[4] A. Santoso and S. T. Gunawan Ariyanto, "Implementasi deep learning berbasis keras untuk pengenalan wajah," Universitas Muhammadiyah Surakarta, 2018.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Adv. Neural Inf. Process. Syst.*, 2012, doi: http://dx.doi.org/10.1016/j.protcy.2014.09.007.

[6] L. Wang and Y. Sun, "Image classification using convolutional neural network with wavelet domain inputs," *IET Image Process.*, 2022.

[7] J.-W. Liu, F.-L. Zuo, Y.-X. Guo, T.-Y. Li, and J.-M. Chen, "Research on improved wavelet convolutional wavelet neural networks," *Appl. Intell.*, vol. 51, no. 6, pp. 4106–4126, 2021.

[8] S. Gunasekaran, S. Rajan, L. Moses, S. Vikram, M. Subalakshmi, and B. Shudhersini, "Wavelet based CNN for diagnosis of COVID 19 using chest X ray," in *IOP Conference Series: Materials Science and Engineering*, 2021, vol. 1084, no. 1, p. 12015.

[9] S. Fujieda, K. Takayama, and T. Hachisuka, "Wavelet convolutional neural networks for texture classification," *arXiv Prepr. arXiv1707.07394*, 2017.

[10] W. Shang, K. Sohn, D. Almeida, and H. Lee, "Understanding and improving convolutional neural networks via concatenated rectified linear units," in *international conference on machine learning*, 2016, pp. 2217–2225.

Image Recognition Menggunakan Metode Cosine Distance untuk Aplikasi Penanganan Food Waste

Monica Chandra[1], Edwin Pramana[1]
[1]Departemen Informatika, Fakultas Sains dan Teknologi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Indonesia

Corresponding author: Penulis A. Monica Chandra (e-mail: monica1@mhs.stts.ac.id).

ABSTRACT The Food and Agriculture Organization of the United Nations (FAO) claimed that 33% - 50% of the food that has been produced is not consumed properly. Also, 11% of food products purchased are wasted, and even if they are not being opened. In 2016-2017, Indonesia itself has become the second largest country after Saudi Arabia, which produced the most food waste in the world. These can have a bad impact on our environment. Therefore, the "Jangan Dibuang" application was made with the aim of reducing the food waste produced. This app was developed for the Android platform with Flutter framework and Amazon Web Service Aurora Database. The "Jangan Dibuang" application is also equipped with image recognition that uses Tensorflow to simplify food searching with an image, which later the image's feature is extracted into matrix and being compared with Cosine Distance method. The "Jangan Dibuang" application can be used by three types of actors, namely administrators, food providers, and buyers. The testing involved seven food providers and 20 buyers. Based on the results of the testing, 201 transactions were made, which saved 285 products. 59 of 201 transactions were aimed at donations. The functionality of the food providers' application gets a score of 79.98% for excellent criteria. For the functionality of the buyers' application, the value obtained is 83% for excellent criteria. For the Image Recognition itself shows the accuracy of 93% after using EfficientNetV2 Keras Application Model which helps to recognize two pictures with different lighting and angle.

KEYWORDS Amazon Web Services, Aurora, Flutter, Food Waste, Tensorflow

ABSTRAK Badan Pangan PBB (FAO) menyatakan 33% - 50% makanan yang telah diproduksi, tidak dikonsumsi dengan semestinya. Selain itu, 11% produk makanan yang dibeli terbuang bahkan tidak dibuka. Tahun 2016-2017, Indonesia sendiri telah menjadi negara terbesar kedua setelah Arab Saudi yang menghasilkan food waste terbanyak di dunia. Penumpukan limbah ini berdampak pada lingkungan. Oleh karena itu, aplikasi "Jangan Dibuang" dibuat dengan tujuan untuk mengurangi food waste yang dihasilkan. Aplikasi ini dibuat untuk platform Android dengan framework Flutter dan database Amazon Web Service Aurora. Selain itu, aplikasi ini juga dilengkapi dengan fitur image recognition yang memanfaatkan Tensorflow untuk mempermudah pencarian makanan dengan sebuah gambar yang mana gambar tersebut akan diekstrak fiturnya menjadi matriks yang kemudian dibandingkan dengan metode Cosine Distance. Aplikasi "Jangan Dibuang" dapat digunakan oleh 3 jenis aktor, yaitu administrator, penyedia makanan, dan pembeli. Uji coba dilakukan terhadap 7 penyedia makanan dan 20 pembeli. Berdasarkan hasil uji coba yang telah dilakukan, didapatkan 201 transaksi, yang mana telah menyelamatkan 285 limbah makanan. 59 dari 201 transaksi ditujukan untuk donasi. Fungsionalitas aplikasi penyedia makanan mendapatkan nilai 79,98% untuk kriteria sangat baik. Untuk fungsionalitas aplikasi pembeli, nilai yang didapatkan adalah 83% untuk kriteria sangat baik. Dari sisi Image Recognition sendiri menunjukkan akurasi 93,3% setelah menggunakan Keras Application Model EfficientNetV2 yang membantu mengenali kedua gambar walaupun dengan pencahayaan dan posisi pengambilan yang berbeda.

KATA KUNCI Amazon Web Services, Aurora, Flutter, Food Waste, Tensorflow

I. PENDAHULUAN

Salah satu permasalahan yang sedang dihadapi dunia saat ini adalah limbah. Limbah terdiri dari berbagai jenis, salah satunya limbah makanan atau yang lebih dikenal sebagai *food waste*. Badan Pangan PBB (Food and Agriculture Organization of The United Nations/FAO) menyatakan bahwa 33% hingga 50% makanan yang telah diproduksi, tidak dikonsumsi dengan semestinya. Hal ini ditunjukan dengan adanya penelitian yang menyatakan bahwa sebanyak 11% produk makanan yang dibeli terbuang bahkan tidak dibuka [1].

Beberapa toko roti telah menerapkan langkah positif untuk mengurangi limbah pangan ini. Alih-alih membuang sisa roti yang tidak terjual pada hari tersebut, toko roti ini memiliki campaign menjual sisa roti tersebut separuh harga pada hari tertentu dan jam tertentu. Oleh karena itu, dengan campaign yang telah dilakukan, hal ini dapat mengurangi limbah makanan sekaligus menekan kerugian toko atas roti-roti yang tidak terjual. Sayangnya, pembelian tersebut hanya dapat dilakukan on the spot. Artinya, pembeli harus datang di toko dan memilih roti yang tersisa. Hal ini dirasa kurang efektif, mengingat setiap orang memiliki kesibukannya masing-masing dan belum tentu memiliki waktu untuk datang on the spot. Oleh karena itu, pembelian food waste seperti ini dapat dilakukan melalui aplikasi.

Mengingat penduduk Indonesia yang mayoritas merupakan pengguna Android, yakni 41 juta pengguna atau pangsa pasarnya sebesar 94%, maka aplikasi "Jangan Dibuang" akan dibuat untuk platform Android. Banyak framework yang mendukung developer untuk mengembangkan aplikasi Android, salah satu yang sedang popular saat ini adalah Flutter yang menawarkan fitur cross-platform dan tampilan yang fleksibel dan menarik. Selain itu, tak dapat dipungkiri, bahwa kecerdasan buatan telah menjangkau hampir seluruh aspek kehidupan manusia. Oleh karena itu, untuk menerapkannya, "Jangan Dibuang" juga dilengkapi dengan fitur Image Recognition untuk memudahkan pembeli untuk mencari makanan sejenis dengan cara memfoto makanan yang dimilikinya serta memudahkan penjual agar mereka tidak perlu menginputkan makanan dari awal lagi setiap harinya, melainkan hanya mengupdate stok setelah aplikasi menemukan makanan yang sesuai dengan hasil Image Recognition. Pada kasus ini, pencarian makanan yang sesuai dilakukan dengan mengekstraksi fitur-fitur pada sebuah gambar dan membandingkannya dengan fitur yang telah tersimpan sebelumnya. Pengukuran kesamaan fitur dilakukan menggunakan Cosine Distance dikarenakan tingkat efisiensinya [2].

Aplikasi "Jangan Dibuang" dibuat agar lebih banyak restoran, bakery, dan penyedia makanan lainnya lebih *aware* dengan food waste dengan cara menjual sisa makanannya melalui aplikasi ini. Selain mengurangi food waste dan meminimalisir kerugian pemilik brand, aplikasi ini juga bermanfaat bagi calon "pahlawan penyelamat" sisa makanan yang hampir terbuang dengan harga miring. Dengan harga miring, setidaknya beban finansial calon pembeli juga berkurang, terutama bagi anak kos dan orang-orang baik yang hendak berbagi makanan/bakti sosial.

II. TINJAUAN PUSTAKA

Flutter merupakan teknologi milik Google yang dapat digunakan untuk membangun aplikasi dengan tampilan UI yang apik. Selain itu, keunggulan Flutter adalah source code yang dibuat dapat di-compile secara native ke dalam aplikasi mobile, web, dan desktop hanya dari satu basis kode. Flutter menggunakan bahasa Dart [11], sebuah bahasa pemrograman yang dikembangkan oleh Google. Amazon Web Services (AWS) adalah platform cloud paling komprehensif dan digunakan secara luas di dunia. Hingga saat ini, AWS menawarkan lebih dari 200 layanan unggulan yang lengkap dari pusat data secara global, salah satunya adalah Aurora. Amazon Aurora adalah basis data relasional yang kompatibel dengan MySQL dan PostgreSQL [7]. Database ini dibangun AWS untuk cloud, yang menggabungkan kinerja dan ketersediaan basis data perusahaan tradisional dengan kesederhanaan dan keefektifan biaya basis data sumber terbuka.

Firebase Cloud Messaging (FCM) dulu dikenal sebagai Google Cloud Messaging (GCM). Firebase Cloud Messaging (FCM) dapat mengirim dan menerima pesan.. Selain tools yang telah disebutkan, digunakan juga Xendit dan Web Service. Xendit adalah perusahaan fintek Indonesia yang menyediakan infrastruktur pembayaran untuk Indonesia. Xendit membantu marketplace mengirimkan pembayaran, pinjaman, dan mendeteksi penipuan. Web service yang merupakan aplikasi yang berisi sekumpulan basis data dan perangkat lunak atau bagian dari program perangkat lunak yang diakses secara remote oleh piranti dengan perantara tertentu. Web service mampu menukar data tanpa memandang sumber database, bahasa yang digunakan, dan pada platform apa data tersebut dikonsumsi. Laravel menyediakan fitur dasar untuk membangun fullstack application, yang menangani request, routing, controller, service, domain/model, hingga view. Namun, Laravel bisa juga dibangun sebagai web service RESTful API.

Image Recognition adalah proses identifikasi dan pendeteksian sebuah objek atau fitur di dalam sebuah gambar digital atau video. Tensorflow adalah library perangkat lunak open source untuk Machine Learning [14]. TensorFlow dapat digunakan dalam berbagai tugas tetapi memiliki fokus utama pada training Neural Network. Tensorflow banyak digunakan dalam kasus-kasus Machine Learning, seperti face recognition untuk mengenali wajah orang pada log absen perkantoran atau dipadukan dengan Embedded System.

Selain penggunaan Tensorflow, untuk memilih fitur-fitur yang digunakan, digunakan Principal Component Analysis. Pada dasarnya analisis komponen utama (PCA) [13] bertujuan menerangkan struktur varians-kovarians melalui

kombinasi linear dari variabel-variabel. Secara umum analisis komponen utama bertujuan untuk mereduksi data dan menginterpretasikannya. Meskipun dari p buah variabel asal dapat diturunkan menjadi p buah komponen utama untuk menerangkan keragaman total sistem (p buah variabel), namun seringkali keragaman total itu dapat diterangkan secara memuaskan oleh sejumlah kecil komponen utama, misal, oleh k buah komponen utama, dimana k < p (k lebih kecil dari pada p). Dalam hal ini, k buah komponen utama dapat menggantikan p buah variabel asal. Analisis komponen utama sering kali bukan merupakan akhir dari suatu pengolahan data tetapi juga merupakan tahap (langkah) antara dalam kebanyakan penelitian yang bersifat lebih luas.

Distance Similarity digunakan untuk menentukan kesamaan antara dokumen atau vektor [6]. Secara matematis, hal ini digunakan untuk mengukur cosine sudut antara dua vektor yang diproyeksikan dalam ruang multi-dimensi. Hubungan antara Distance Similarity dan Cosine Distance berkebalikan. Cosine Distance ahkan menurun apabila Similarity tinggi. Rumus untuk Cosine Distance dapat dilihat pada formula 1 [9].

$$Cosine\ Distance = 1 - \frac{u.v}{||u||||v||} \quad (1)$$

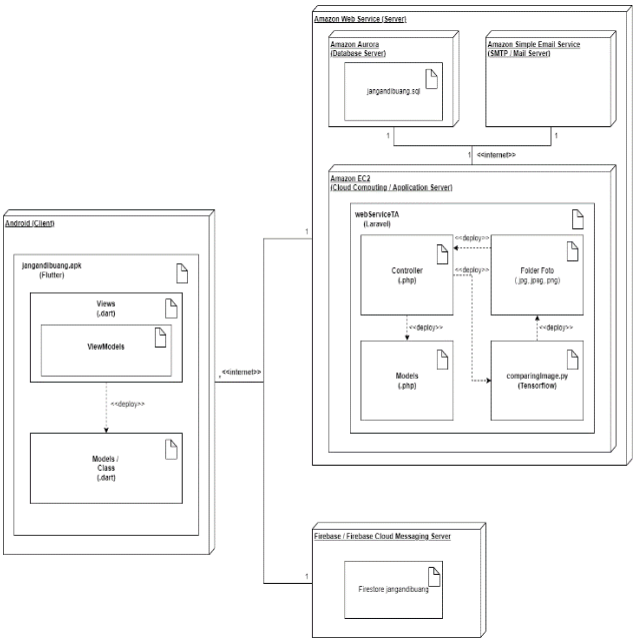
Penelitian pengenalan makanan menggunakan Image Recognition telah banyak dilakukan sebelumnya. Kong et al. [3] pernah membuat aplikasi mobile bernama DietCam. DietCam menggunakan algoritma SIFT feature descriptor dan Nearest Neighbour untuk mengenali makanan. Melalui penelitian ini, akurasi yang didapatkan sebesar 92%. Andrews Samraj et al. [4] dalam artikelnya yang berjudul *"Food Genre Classification from Food Images by Deep Neural Network with Tensorflow and Keras"* juga pernah melakukan proses klasifikasi jenis makanan dengan Neural Network memanfaatkan Tensorflow dan Keras. Andrews bersama timnya menggunakan dataset berisi 170 gambar makanan dalam bentuk array berdimensi (350, 350, 3). Untuk meningkatkan akurasi dan mengurangi *loss* pada model, tim Andrew menggunakan teknik Back Propagation Multilabel Learning (BP MLL). Dalam pengerjaannya, *library* Keras digunakan untuk back propagation dan Tensorflow digunakan untuk mengasosiasikan data point dengan sebuah set berisi label-label. Penelitian lainnya yang masih terkait dengan makanan dilakukan oleh Nupur Bhave dan Dipti Belsare [5] untuk mengenali makanan beserta estimasi kandungan nutrisi pada makanan tersebut. Peneliti menggunakan deep learning dengan metode Convolutional Neural Networks (CNN). Pengaplikasian CNN dilakukan dengan menggunakan Tensorflow dan Keras.

III. *RANCANGAN SISTEM*

Pada bagian ini akan dijelaskan mengenai rancangan sistem, yakni rancangan arsitektur mencakup alur image recognition, serta use case aktor-aktor yang terlibat.

A. ARSITEKTUR SISTEM

Bagian A menjelaskan arsitektur sistem yang digunakan dalam aplikasi. Arsitektur yang akan dibahas adalah dituangkan dalam bentuk Deployment Diagram agar setiap komponen dapat dijelaskan lebih rinci.



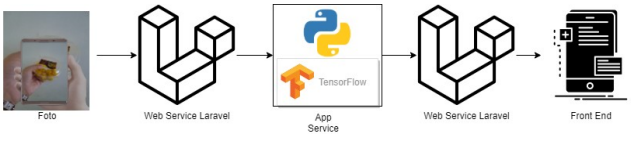
GAMBAR 1. Arsitektur Sistem Aplikasi.

Gambar 1 menunjukkan arsitektur sistem yang dibuat. Aplikasi yang dibuat berbasiskan Android dengan menggunakan framework Flutter. Framework Flutter memiliki arsitektur Model-View-View Model atau disingkat sebagai MVVM. Data yang ditampilkan diambil dari web service berbasis Laravel yang terhosting di Amazon Web Service (AWS). Dalam hal ini, View Model yang bertugas sebagai mediator pengambilan data. Pengambilan data memerlukan koneksi internet untuk menghubungkan node client dengan node AWS. Seperti yang digambarkan, hubungan asosiasi kedua node memiliki derajat keserbaragaman 1 untuk AWS dan banyak (*) untuk client. Kemudian data-data yang diperoleh akan ditampung dalam array list of object. Dalam hal ini, model/ class yang berperan. Terakhir, data-data tersebut ditampilkan pada view. Oleh karena itu, view memiliki dependensi ke model, sehingga dilambangkan dengan <<deploy>>.

Aplikasi memanfaatkan beberapa service dari Amazon Web Service (AWS). Beberapa service tersebut di antaranya Amazon Aurora sebagai database server, Amazon Simple Mail Service sebagai mail server, dan Amazon EC2 sebagai application server. Amazon EC2 memiliki asosiasi dengan Amazon Aurora dan Amazon Simple Mail Service, masing-masing dengan derajat keserbaragaman 1 dengan 1. Amazon Aurora merupakan Relational Database Service yang mendukung penggunaan PostgreSQL dan MySQL. Dalam aplikasi ini digunakan MySQL, yakni database jangandibuang.sql.

Aplikasi menggunakan web service berbasis Laravel. Laravel memiliki arsitektur Model-View-Controller (MVC)[10]. Namun, karena hanya digunakan sebagai web service, komponen yang digunakan hanya model dan controller. Beberapa fungsi pada controller memiliki kemampuan untuk menyimpan gambar ke dalam folder foto. Oleh karena itu, folder foto memiliki dependensi ke controller.

Aplikasi dilengkapi dengan fitur image recognition yang digunakan untuk mencari produk sejenis bagi pembeli dan memberikan rekomendasi daily waste yang akan ditambahkan bagi penyedia makanan. Fitur image recognition ini memanfaatkan Tensorflow. Dalam hal ini, file yang akan dijalankan adalah comparingImage.py. Secara garis besar, proses dimulai ketika ada makanan yang difoto. Foto tersebut kemudian dibandingkan dengan foto-foto yang tersimpan. Sebagai hasil, dikembalikan rekomendasi list makanan yang sesuai dengan makanan yang difoto oleh user.



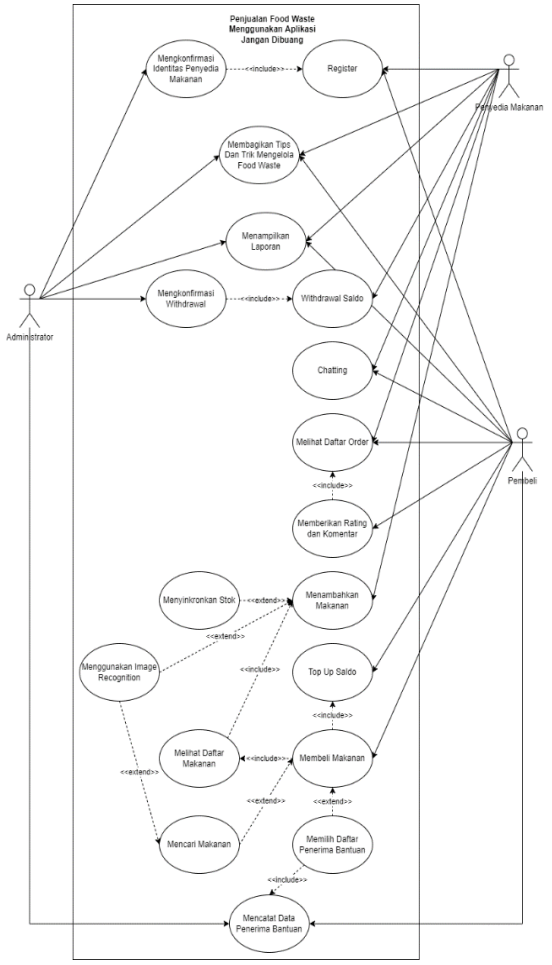
GAMBAR 2. Alur Image Recognition.

Gambar 2 menunjukkan alur image recognition. Menurut alurnya, proses dimulai dengan user memfoto makanan menggunakan smartphonenya. Kemudian gambar ini akan diterima oleh web service Laravel. Laravel akan menerima foto tersebut, menyimpannya, mengirimkan nama file, kemudian mengeksekusi file python. File python akan mengambil seluruh foto yang tersimpan dan mencari foto yang memiliki kemiripan dengan foto yang baru saja dikirimkan oleh user. Setelah proses selesai, hasilnya akan dikembalikan ke Laravel dan Laravel mengembalikan rekomendasi list makanan-makanan terkait ke front end. Oleh karena itu, file comparingImage.py memiliki dependensi terhadap folder foto.

Aplikasi juga dilengkapi dengan fitur chatting antara penyedia makanan dengan pembeli. Sesuai dengan gambar 1, fitur chatting ini memanfaatkan fitur dari Firebase, yakni Firebase Cloud Messaging. Daftar chatting dan user tersimpan di dalam firestore jangandibuang. Node Android (client) dan FCM server dihubungkan dengan asosiasi yang memiliki derajat keserbaragaman 1 (FCM server) dengan banyak (*) untuk client.

B. USE CASE DIAGRAM

Aplikasi memiliki 3 (tiga) aktor yang terlibat. 3 aktor ini meliputi administrator, penyedia makanan, dan pembeli. Setiap jenis aktor memiliki hak akses yang berbeda-beda. Berikut merupakan use case untuk menjelaskan hubungan antara aktor dan sistem.



GAMBAR 3. Alur Image Recognition.

Mula-mula, penyedia makanan dan pembeli melakukan registrasi. Khusus untuk penyedia makanan, akunnya perlu dikonfirmasi oleh administrator sehingga jelas dan dapat dipertanggungjawabkan. Oleh karena itu, pada use case diagram, dihubungkan dengan <<include>>. Setelah melalui proses konfirmasi, barulah penyedia makanan dapat menjual makanannya. Saat pertama kali menginputkan makanan, penyedia makanan harus menginputkan semua data yang diperlukan, termasuk nama, foto, dan harga makanan. Namun, apabila makanan telah diinputkan sebelumnya dan besoknya masih ada sisa makanan dengan jenis yang sama, penyedia makanan tidak perlu memasukkan data makanan dari awal, melainkan menggunakan fitur image recognition untuk mengenali makanan dan memberikan rekomendasi jenis makanan tersebut, sehingga penyedia makanan hanya perlu menginputkan jumlah makanannya saja. Hubungan ini digambarkan dengan <<extend>>.

Pembeli dan penyedia makanan dapat melakukan chatting untuk mempermudah komunikasi. Keduanya juga dapat melihat daftar order masing-masing. Khusus untuk pembeli, setelah order tersebut terselesaikan, pembeli dapat memberikan rating dan komentar. Dalam hal ini, pada use case diagram ditunjukkan dengan hubungan memberikan rating

dan komentar <<include>> melihat daftar order. Pembelian pada aplikasi menggunakan e-wallet/saldo dalam aplikasi. Oleh karena itu, pembeli dapat melakukan top up saldo, sedangkan penyedia makanan dapat melakukan withdrawal saldo. Withdrawal saldo memerlukan konfirmasi dari administrator. Oleh karena itu pada use case diagram, digambarkan mengkonfirmasi withdrawal oleh administrator <<include>> withdrawal saldo.

IV. AMAZON WEB SERVICE AURORA

Secara garis besar, seluruh penyimpanan dilakukan pada AWS Aurora. AWS Aurora merupakan basis data relasional yang kompatibel dengan MySQL dan PostgreSQL yang dibangun untuk cloud, yang menggabungkan kinerja dan ketersediaan basis data perusahaan tradisional dengan kesederhanaan dan keefektifan biaya. Amazon Aurora sepenuhnya dikelola oleh Amazon Relational Database Service (RDS), yang mengotomatiskan tugas administrasi yang memerlukan banyak waktu seperti penyediaan perangkat keras, penyiapan basis data, pengelolaan patch, dan pencadangan.

Database AWS Aurora digunakan untuk kemudahan penggunaan, yakni segala administrasi hingga backup yang dilakukan secara otomatis oleh Amazon sehingga memanfaatkan teknologi serverless dan tidak perlu melakukan administrasi awal secara manual. Alasan kedua, aplikasi ini berjalan di mobile dengan bahasa pemrograman Dart. Di satu sisi, untuk Image Recognition, perlu menggunakan bantuan library Tensorflow yang berjalan di atas bahasa pemrograman Python. Amazon memudahkan keseluruhan rangkaian system, karena menyediakan fitur Virtual Private Cloud (VPC) yang dapat menyimpan keseluruhan sistem, yakni aplikasi, web service, dan lainnya. Selain itu, Amazon juga dapat menjalankan Secure Shell (SSH). Hal ini tentu lebih efisien jika dibandingkan ketika web service dan database dihosting secara terpisah-pisah.

Selain penggunaan AWS, dalam aplikasi ini juga memanfaatkan Firebase. Firebase dalam kasus ini digunakan untuk membantu proses chatting antara pembeli dan penyedia makanan. Fitur yang dimaanfaatkan adalah Firestore. Collection di Firestore menyimpan data berupa daftar chat dan pesan-pesan yang dikirimkan.

V. IMAGE RECOGNITION MENGGUNAKAN TENSORFLOW

Bagian ini menjelaskan konsep yang dimiliki oleh Tensorflow. Tak hanya itu, pada bagian ini dijabarkan pula model yang digunakan dalam pembuatan modul Image Recognition pada aplikasi. Selain itu, dijelaskan cara penggunaan Tensorflow untuk aplikasi. Berikut merupakan penjabaran dari Tensorflow.

A. KERAS

Keras merupakan library Machine Learning open source berbasis Python. Keras dikembangkan untuk membuat penerapan model pembelajaran yang mendalam secepat dan semudah mungkin untuk penelitian serta pengembangan yang dirilis berdasarkan lisensi MIT. Library jaringan saraf yang bersifat open source ini dirancang untuk memberikan eksperimen cepat dengan jaringan saraf yang dalam, dan dapat berjalan di atas CNTK, Tensorflow, dan Theano. Keras berfokus untuk menjadi modular, user friendly, dan bersifat extensible. Kerangka kerja ini tidak menangani komputasi tingkat rendah namun sebaliknya, akan menyerahkan tugas tersebut ke library lain yang disebut backend.

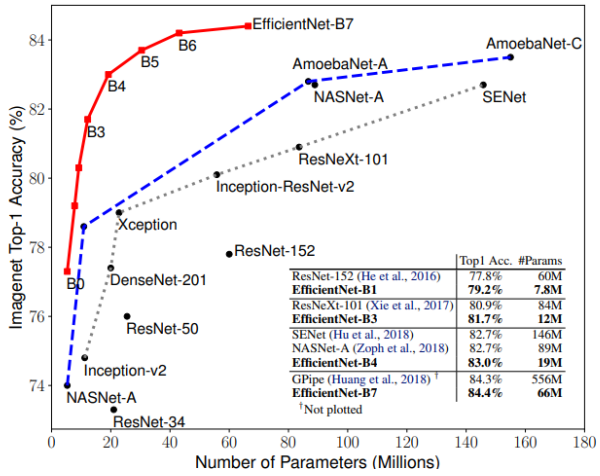
Backend adalah istilah dalam Keras yang melakukan semua perhitungan tingkat rendah. Perhitungan ini seperti produk tensor, konvolusi, dan banyak hal lain dengan bantuan library lain, seperti Tensorflow atau Theano. Jadi, backend akan melakukan perhitungan dan pengembangan model. Tensorflow adalah backend default tetapi pengguna dapat mengubahnya dalam konfigurasi.

Keras API menangani cara membuat model, mendefinisikan layer, atau mengatur beberapa model input-output. Pada level ini, Keras juga mengkompilasi model dan mengoptimalkan, serta mendukung proses pelatihan dengan fungsi fit. Keras diadopsi dan diintegrasikan ke dalam Tensorflow pada pertengahan 2017. Pengguna dapat mengaksesnya melalui modul tf.keras. Namun, library Keras masih dapat beroperasi secara terpisah dan mandiri.

B. KERAS APPLICATIONS

Keras Applications mencakup model yang berasal dari domain deep learning dan memiliki skenario bobot yang telah dilatih sebelumnya. Model aplikasi Keras memiliki penggunaannya dalam spektrum yang luas termasuk fine-tuning, prediksi dan ekstraksi fitur. Ketika model digunakan, maka secara otomatis semua bobot yang diperlukan untuk model secara otomatis diunduh dan disimpan dalam folder /.keras/models. Ketika Keras digunakan secara instan maka sesuai format data gambar, model Keras dibangun dengan mempertimbangkan semua konfigurasi yang disebutkan di dalam file Keras yang terletak di .keras/Keras.JSON.

Arsitektur model Keras sepenuhnya kompatibel dengan backend CNTK, Theano, dan Tensorflow. Beberapa model yang disediakan di antaranya VGG16, VGG19, Xception, InceptionV3, ResNet50, MobileNet, InceptionResNetV2, EfficientNet, EfficientNetV2, DenseNet, NASNet-A, AmoebaNet-A, SENet, dan lain-lain. Masing-masing model memiliki ukuran dan akurasi yang berbeda-beda. Untuk penggunaan setiap application sangat mudah dan mirip untuk semua application. Pengguna cukup memuat salah satu model aplikasi Keras yang diperlukan dengan mengimpor Keras dan model yang diperlukan dari Keras. Alih-alih berupaya untuk melatih model yang telah dibuat sendiri, pengguna dapat menggunakan model yang tersedia yang telah ditentukan sebelumnya dalam aplikasi Keras.



GAMBAR 4. Ukuran Model vs Akurasi ImageNet

Gambar 4 menunjukkan grafik yang membandingkan ukuran model dari Keras Applications dengan akurasi ImageNet. Sumbu x melambangkan jumlah parameter, sedangkan sumbu y melambangkan tingkat akurasi ImageNet. Semakin kiri dari sumbu X, berarti jumlah parameter semakin sedikit dan berarti ukuran model juga semakin kecil. Pada aplikasi "Jangan Dibuang", model yang digunakan adalah EfficientNetV2.

C. KERAS APPLICATIONS EFFICIENTNETV2

Keras Applications mencakup model yang berasal dari domain deep learning dan memiliki skenario bobot yang telah dilatih sebelumnya [12]. Model aplikasi Keras memiliki penggunaannya dalam spektrum yang luas termasuk fine-tuning, prediksi dan ekstraksi fitur. Ketika model digunakan, maka secara otomatis semua bobot yang diperlukan untuk model secara otomatis diunduh dan disimpan dalam folder /.keras/models. Ketika Keras digunakan secara instan maka sesuai format data gambar, model Keras dibangun dengan mempertimbangkan semua konfigurasi yang disebutkan di dalam file Keras yang terletak di .keras/Keras.JSON.

D. PICKLE

Modul Pickle [8] merupakan library Python yang mengimplementasikan protokol biner untuk membuat serialisasi dan de-serialisasi struktur objek. "Pickling" adalah proses di mana hierarki objek Python diubah menjadi aliran byte, dan "unpickling" adalah operasi terbalik, di mana aliran byte (dari file biner atau objek seperti byte) diubah kembali menjadi hierarki objek. Pickling (dan unpickling) juga dikenal sebagai serialisasi, marshalling, atau flattening. Namun, untuk menghindari kebingungan, istilah yang digunakan di sini adalah "pickling" dan "unpickling".

Format data yang digunakan oleh Pickle adalah format khusus Python. Hal ini memiliki keuntungan bahwa tidak ada batasan yang diberlakukan oleh standar eksternal seperti JSON atau XDR (yang tidak dapat mewakili berbagi pointer). Namun, hal ini berarti bahwa program non-Python mungkin tidak dapat merekonstruksi objek Python menggunakan Pickle. Secara default, format data Pickle menggunakan representasi biner.

E. PENERAPAN TENSORFLOW UNTUK IMAGE RECOGNITION

Pada subbab ini dijelaskan mengenai implementasi dari hal-hal yang berkaitan dengan image recognition. Aplikasi Jangan Dibuang menyediakan fitur image recognition untuk membantu pembeli dalam mencari produk sejenis dan penyedia makanan dalam menambahkan daily waste. Image Recognition dijalankan dengan mengeksekusi file python pada web service Laravel. Image Recognition ini memanfaatkan Tensorflow dan menggunakan algoritma Cosine Distance.

Dalam proses image recognition yang dilakukan pada aplikasi "Jangan Dibuang", mula-mula gambar diekstraksi fiturnya menjadi numpy array yang diserialisasi menggunakan Pickle. File Pickle disimpan dengan ekstensi .pkl yang bisa dibaca dan ditulis setiap ada gambar baru yang diinputkan ke dalam folder makanan.. Terdapat 2 (dua) file Pickle. File pertama digunakan untuk menyimpan fitur-fitur, sedangkan file kedua menyimpan nama file gambar-gambar yang nantinya dikembalikan untuk diolah kembali pada Laravel.

Alur Image Recognition dimulai ketika pengguna memfoto sebuah makanan yang ada didepannya melalui kamera yang disediakan pada aplikasi. Setelah gambar diambil, aplikasi akan men-*trigger* web service Laravel untuk dijalankan. Web service Laravel akan mengeksekusi file Python telah terlebih dahulu diletakkan pada folder public dengan perintah *exec("python 3.9 ./compareImage.py {$path_foto}",$output)*. Beberapa library yang dibutuhkan di antaranya pickle, keras.preprocessing, numpy, matplotlib.pyplot, scipy.spatial, os, dan sys. Agar dapat menggunakan library cosine distance, maka numpy array features harus dijadikan 1(satu) dimensi dengan menggunakan perintah ravel.

Mula-mula, gambar yang dikirimkan dari aplikasi akan diekstraksi fiturnya menjadi numpy array. Nilai pada numpy array ini kemudian dibandingkan dengan numpy array lainnya yang telah ditampung pada file pickle. Perhitungan perbandingan ini menggunakan metode Cosine Distance dan mengembalikan gambar-gambar dengan nilai Cosine Distance yang kecil. Hasil dari proses ini berupa nama-nama file gambar serupa yang ditampung ke dalam array output. Array output ini yang kemudian dikembalikan oleh web service ke aplikasi.

VI. HASIL UJI COBA

Bagian ini akan menjelaskan mengenai uji coba yang digunakan pada pembuatan Aplikasi Marketplace Food Waste Berbasis Android Menggunakan Flutter dan Database Amazon Aurora Dilengkapi Image Recognition Memanfaatkan Tensorflow. Uji coba ini mencakup uji coba

yang dilakukan secara black box testing dan kuesioner. Kuesioner dibagikan ke para pengguna untuk menilai kelayakan dan tanggapan aplikasi yang dibuat.

Uji coba dilakukan terhadap 7 penyedia makanan dan 20 pembeli. Berdasarkan hasil uji coba yang telah dilakukan, didapatkan 201 transaksi penjualan food waste, yang mana telah menyelamatkan 285 limbah makanan. 59 dari 201 transaksi ditujukan untuk donasi. Fungsionalitas aplikasi penyedia makanan mendapatkan nilai 79,98% untuk kriteria sangat baik. Untuk fungsionalitas aplikasi pembeli, nilai yang didapatkan adalah 83% untuk kriteria sangat baik. 7 dari 7 penyedia makanan yang mengikuti uji coba bersedia bergabung. Begitu pula dengan pembeli, 20 dari 20 penguji coba bersedia menggunakan aplikasi ketika aplikasi dirilis.

GAMBAR 5. Hasil Image Recognition

Tak hanya fitur transaksi penjualan food waste secara keseluruhan, namun Gambar 5 juga menunjukkan hasil uji coba Image Recognition pada aplikasi yang dibuat. Gambar sebelah kiri merupakan foto yang diambil pada saat program dijalankan. Setelah program dijalankan, program mampu mengembalikan data makanan sejenis, yakni makanan dengan gambar sebelah kanan yang telah tersimpan pada database sebelumnya. Dari gambar di atas, dapat dibuktikan bahwa program telah berhasil mengenali makanan sejenis walaupun kedua gambar memiliki pencahayaan ataupun posisi pengambilan yang berbeda.

Dari 60 (enam puluh) percobaan yang dilakukan, 56 percobaan menunjukkan hasil yang benar. Hal ini berarti, penelitian ini memberikan akurasi senilai 93,3%. Dengan menggunakan Tensorflow, Cosine Distance, dan Keras Applications EfficientNetV2, akurasi dapat ditingkatkan. Hal ini dibuktikan dengan dapat dikenalinya suatu makanan walaupun pencahayaan dan posisi pengambilannya berbeda. Sebagai informasi, percobaan yang dilakukan telah melewati proses pembandingan kedua gambar yang memang betul-betul sama, kedua gambar dengan pencahayaan berbeda, kedua gambar dengan posisi pengambilan berbeda, dan kedua gambar yang memang sangat berbeda.

VII. KESIMPULAN

Bagian ini akan membahas mengenai kesimpulan dan saran mengenai aplikasi ini. Berikut merupakan kesimpulan yang didapatkan:

1. Aplikasi Jangan Dibuang telah menjadi sebuah aplikasi mobile yang mendukung orang-orang untuk saling berbagi makanan kepada mereka yang membutuhkan. Hal ini dibuktikan dari hasil uji coba, 59 transaksi ditujukan untuk tujuan donasi ke beberapa Panti Asuhan yang telah disediakan daftarnya oleh administrator. Selain itu, dalam pembuatannya, aplikasi mengimplementasikan framework Flutter dan database Amazon Aurora. Penggunaan framework Flutter memberikan kemudahan dalam pengaturan tampilan. Salah satunya, apabila sebelumnya pembuatan List View pada proses development *Android Native* menggunakan Java harus dibuat dalam class terpisah, dengan adanya framework Flutter, pembuatan List View lebih mudah karena komponennya telah tersedia dengan bantuan library-library yang dimiliki oleh Flutter. Untuk database Amazon Aurora, Aurora memiliki keunggulan dari sisi ketersediaan dan reliabilitasnya. Hal ini terbukti dengan adanya fitur replikasi otomatis yang telah disediakan dan waktu pemulihan kurang dari satu menit.
2. Aplikasi telah membantu mengurangi food waste yang dihasilkan penyedia makanan dan menekan beban finansial calon pembeli dengan penjualan harga miring. Dengan adanya penjualan harga miring, lebih banyak makanan yang terselamatkan. Apabila sebelumnya penyedia makanan memilih untuk membuang sisa makanan, melalui aplikasi, sisa makanan yang masih layak konsumsi dapat diperjualbelikan. Sebagai contoh, 285 produk telah terselamatkan dalam kurun waktu uji coba dengan harga hampir dari setengah harga asli. Selain itu, 83% pembeli juga puas terhadap fungsionalitas dari aplikasi, serta 20 dari 20 pembeli ingin menggunakan aplikasi ketika nantinya aplikasi dirilis. Selain itu, dari sisi Image Recognition menggunakan Keras Application EfficientNetV2 memiliki tingkat akurasi lebih tinggi dibandingkan dengan penggunaan source code dan algoritma yang sama namun memanfaatkan model VGG16. Cosine Distance juga meningkatkan efisiensi perhitungan karena hasilnya selalu ternormalisasi.

PERAN PENULIS

Monica Chandra: Tinjauan Pustaka, Rancangan dan Implementasi Sistem, Ujicoba.
Edwin Pramana: Konseptual, Penyuntingan

COPYRIGHT

DAFTAR PUSTAKA

[1] Wansink, B., "Abandoned Products and Consumer Waste: How Did That Get Into the Pantry?," Choices, 16(316-2016-6512), 2001.
[2] Baoli Li and Li Ping Han, "Distance Weighted Cosine Similarity Measure for Text Classification," Henan University of Technology, 2013, DOI: 10.1007/978-3-642-41278-3_74
[3] Kong F and Tan J, "Dietcam: Automatic dietary assessment with mobile camera phones," Pervasive Mob Comput 8(1), pp 147–163, 2012.
[4] Andrews Samraj et al., "Food Genre Classification from Food Images by Deep Neural Network with Tensorflow and Keras," 2020.
[5] Nupur Bhave and Dipti Belsare, "A Study on Food Recognition & Nutrition Estimation," International Peer Reviewed Journal, ISSN 0973-2861 Vol XVI, Issue IV, Jan-June 2022.
[6] Anjani Kumar. 2020. Cosine Similarity & Cosine Distance.[Online]. Available: https://medium.datadriveninvestor.com/cosine-similarity-cosine-distance-6571387f9bf8.
[7] Anonim. Amazon Aurora.[Online]. Available: https://aws.amazon.com/id/rds/aurora/.
[8] Anonim. Pickle — Python Object Serialization [Online]. Available at : https://docs.python.org/3/library/pickle.html.
[9] Anonim. Scipy Spatial Cosine Distance API Reference.[Online]. Available: https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.distance.cosine.html.
[10] Kevin NFA. 2020. Laravel-Pengertian, Kelebihan, Kekurangan dan Cara Install Laravel. [Online]. Available: https://medium.com/@kevinnfa0107/laravel-pengertian-kelebihan-kekurangan-dan-cara-install-laravel-224a79550a91.
[11] Muhammad Amirul Ihsan. 2020. Apa Itu Dart?.[Online]. Available: https://www.kawankoding.id/apa-itu-dart/.
[12] Mostafa Ibrahim. 2021. Google releases EfficientNetV2 — a smaller, faster, and better EfficientNet. [Online]. Available at: https://towardsdatascience.com/google-releases-efficientnetv2-a-smaller-faster-and-better-efficientnet673a77bdd43c#:~:text=EfficientNetV2%20uses%20the%20concept%20of%20progressive%20learning%20which,speeds%20start%20to%20suffer%20on%20high%20image%20sizes.
[13] Richie. 2022. Principal Component Analysis (PCA).[Online]. Available: https://www.mobilestatistik.com/principal-component-analysis-pca/.
[14] Wede. 2020. Belajar Data Science : Apa yang dimaksud dengan Tensorflow dan Bagaimana Penggunaannya?.[Online]. Available: https://dqlab.id/belajar-data-science-pahami-tensflow.

Pembentukan Aturan Fuzzy Untuk Pemberian Rekomendasi Penerima Bantuan Keluarga Berumah Tidak Layak Huni Menggunakan K-means Clustering

Aidil[1], Judi Prajetno[2], Esther Irawati Setiawan[3], and Adi Surya Putra[4]
[1]Departemen Informatika, STMIK PPKIA Tarakanita Rahmawati, Tarakan, Indonesia
[2]Departemen Elektro, Fakultas Sains dan Teknologi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Indonesia
[3]Departemen Teknologi Informasi, Fakultas Sains dan Teknologi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Indonesia
[4]Departemen Sistem Informasi, Fakultas Sains dan Teknologi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Indonesia

Corresponding author: Esther Irawati Setiawan (e-mail: esther@istts.ac.id).

ABSTRACT Assistance for families whose houses are not livable is one of the social benefits given to families who are experiencing financial difficulties and/or have houses that are not livable. The variables considered when determining beneficiaries often make decisions difficult to make. Therefore, we need a fuzzy reasoning system that automatically generates rules as the expected decision makers. To form fuzzy rules, an expert is needed. An expert is a person who is experienced and able to explain a rule related to a field. In this research, a rule is formed automatically which does not depend on an expert. The fuzzy rules generated can be obtained from several techniques such as the clustering process. The method used in generating this fuzzy rule is the k-means clustering method. K-means clustering is used to group data and generate rules in the case of recommendations for recipients of uninhabitable housing. The results of the generation of fuzzy rules are used for the fuzzy inference process using the Sugeno Fuzzy Inference System method. The Sugeno method produces an output (consequently) in the form of a constant or a linear equation. In this study, 1000 training data were used and a process of testing 300 test data was carried out to obtain recommendations for recipients of uninhabitable housing assistance. The test results are used to assess the accuracy of the rules developed. The study's findings reveal that the results of k-means clustering may automatically produce rules for the production of Sugeno Fuzzy Inference System rules in more than 75% of cases.

KEYWORDS Clustering, Fuzzy, Fuzzy Inference System, Fuzzy Rules, K-Means, Sugeno, Uninhabitable Houses

ABSTRAK Bantuan bagi keluarga yang rumah tidak layak huni merupakan salah satu manfaat sosial yang diberikan kepada keluarga yang mengalami kesulitan keuangan dan/atau memiliki rumah tidak layak huni. Variabel yang dipertimbangkan saat menentukan penerima manfaat sering kali membuat keputusan sulit diambil. Oleh karena itu, diperlukan sistem penalaran fuzzy yang secara otomatis menghasilkan aturan-aturan sebagai pembuat keputusan yang diharapkan. Untuk membentuk aturan fuzzy diperlukan seorang pakar. Pakar adalah seorang ahli yang berpengalaman dalam suatu bidang yang mampu menjelaskan suatu aturan yang terkait dengan suatu bidang. Dalam penelitian ini dibentuk suatu rule secara otomatis yang tidak tergantung dengan seorang pakar. Aturan fuzzy dibangkitkan bisa diperoleh dari beberapa teknik seperti proses clustering. Metode yang digunakan dalam membangkitkan aturan fuzzy ini yaitu metode k-means clustering. Dalam hal rekomendasi penerima bantuan rumah tidak layak huni, K-means clustering digunakan untuk mengelompokkan data dan mengembangkan aturan. Hasil dari pembangkitan aturan fuzzy digunakan

untuk proses inferensi fuzzy menggunakan metode Fuzzy Inference System Sugeno. Metode sugeno menghasilkan output (konsekuen) berupa konstanta atau persamaan linier. Dalam penelitian ini digunakan 1000 data training dan dilakukan proses pengujian 300 data uji untuk mendapatkan rekomendasi penerima bantuan rumah tidak layak huni. Hasil pengujian digunakan untuk mengetahui akurasi aturan yang terbentuk.Dari hasil penelitian menunjukkan bahwa hasil k- means clustering dapat membentuk rule secara otomatis untuk pembangkitan aturan Fuzzy Inference System Sugeno dapat dilihat dari hasil akurasi perhitungan pengujian data uji skenario global sama-sama menghasilkan akurasi minimal di atas 75%.

KATA KUNCI Aturan Fuzzy, Clustering, Fuzzy, Fuzzy Inference System, K-Means, Rumah Tidak Layak Huni, Sugeno

I. PENDAHULUAN

Bantuan keluarga berumah tidak layak huni adalah salah satu bantuan sosial yang dihibahkan kepada keluarga yang menghadapi kesulitan ekonomi dan atau mempunyai rumah yang tidak layak huni [1]. Terdapat berbagai variabel-variabel yang perlu diperhatikan dalam penentuan penerima bantuan menyebabkan kesulitan untuk mengambil keputusan. Untuk itulah diperlukan suatu sistem otomatis inferensi Fuzzy yang membangkitkan aturan sebagai pengambil keputusan.

Dari tantangan ini, muncul ide pemikiran untuk mengembangkan suatu penelitian berbasis fuzzy yang menghasilkan aturan-aturan fuzzy secara otomatis. Untuk membentuk aturan fuzzy diperlukan seorang pakar. Pakar adalah seorang ahli yang berpengalaman dalam suatu bidang yang mampu menjelaskan suatu aturan yang terkait dengan suatu bidang .

Dalam penelitian ini dibentuk suatu rule secara otomatis yang tidak tergantung dengan seorang pakar. Aturan fuzzy dibangkitkan bisa diperoleh dari beberapa teknik seperti proses clustering. Metode yang digunakan dalam membangkitkan aturan fuzzy ini yaitu metode k-means clustering. K-means clustering digunakan untuk mengelompokkan data dan membangkitkan aturan pada kasus rekomendasi penerima bantuan keluarga berumah tidak layak huni. Hasil dari pembangkitan aturan fuzzy digunakan untuk memproses inferensi fuzzy.

Langkah yang diambil adalah membuat aplikasi pendukung keputusan yang diharapkan dapat mempermudah untuk melakukan seleksi rekomendasi penerima bantuan keluarga berumah tidak layak huni di dinas sosial dan tenaga kerja kota Tarakan. Dalam makalah ini, penentuan penerimaan bantuan keluarga berumah berdasarkan beberapa kriteria diantaranya umur, penghasilan, jumlah tanggungan (orang) dan kondisi rumah (lantai, dinding, atap dan WC).

Selain itu, penelitian yang diusulkan hanya menentukan seleksi rekomendasi penerima bantuan berumah tidak layak huni tanpa menentukan nilai nominal bantuannya. Sedangkan data yang digunakan pada proses pelatihan adalah data keluarga berumah tidak layak huni tahun 2012 sebanyak 500 data dan 2013 sebanyak 500 data. Data yang digunakan untuk pengujian sistem adalah data keluarga berumah tidak layak huni 2014 sebanyak 300 data. Data yang di ambil sebagai data uji dari kota Tarakan.

Pada makalah ini, clustering data diproses menggunakan K-means dengan analisis varian untuk menentukan cluster terbaik dan pengujian datanya menggunakan sistem inferensi fuzzy model Sugeno. Makalah ini bertujuan untuk mengimplementasikan sistem yang dapat memberikan rekomendasi penerima bantuan keluarga berumah tidak layak huni. Selain itu, diharapkan penelitian ini dapat mempermudah dan mempercepat dalam proses pencarian keluarga yang direkomendasikan mendapat bantuan keluarga berumah tidak layak huni berdasarkan kebutuhan kuota penerima.

Target lainnya yang ingin dicapai adalah meminimalisir berkurangnya kuota penerima bantuan berumah tidak layak huni di tahun berikutnya karena kesalahan dalam seleksi pemilihan. Hasil penelitian ini akan dibandingkan dengan hasil seleksi pemberian bantuan keluarga berumah tidak layak huni yang dilaksanakan oleh Departemen Sosial dan Tenaga Kerja

Kontribusi penelitian ini adalah :

1. Membantu para pegawai dinas sosial dan tenaga kerja untuk menentukan pemberian bantuan keluarga berumah tidak layak huni sesuai dengan kriteria yang di inginkan.
2. Mengetahui sejauh mana kegunaan Fuzzy dalam kehidupan sehari – hari.
3. Sampai dengan saat ini, masih belum ada program aplikasi yang dapat membantu dalam penunjang pengambilan keputusan untuk memberikan bantuan keluarga berumah tidak layak huni.

II. TINJAUAN PUSTAKA

Beberapa hal yang menjadi dasar teori dari penelitian ini yaitu teori program bedah rumah (Rumah Tidak Layak Huni), data mining, k-means clustering, analisa varian, logika fuzzy, fuzzy inference system sugeno.

A. PROGRAM BEDAH RUMAH (RUMAH TIDAK LAYAK HUNI)

Rumah itu adalah kebutuhan manusia mendasar yang berfungsi sebagai lokasi untuk hidup atau dihuni serta metode membesarkan keluarga. Pada dasarnya, setiap individu membutuhkan rumah yang layak, tetapi bagi sebagian orang, memenuhi permintaan untuk perumahan yang layak huni adalah kesulitan.

Selain itu, Rumah adalah salah satu indikator kemakmuran. Kondisi rumah yang layak dihuni mencerminkan kesejahteraan yang tinggal di rumah, sementara kehadiran banyak rumah komunitas yang tidak dapat dihuni menunjukkan bahwa orang-orang di wilayah tersebut tidak kaya. Pemerintah Kota Tarakan secara resmi mempresentasikan hasil program kepada pemilik rumah pada 2012, setelah menyelesaikan program bedah rumah atau rehabilitasi rumah tidak layak dihuni (RTLH).

Layanan Sosial Kota Tarakan, bekerja sama dengan pihak ketiga, meluncurkan program renovasi DPR pada 30 Oktober 2012, dengan target 100 RTLH yang tersebar di empat distrik, yaitu Distrik Tarakan Barat, Tarakan Utara, Tarakan Tengah, dan Tarakan Timur, yang memiliki sejak diperpanjang selama beberapa tahun [2].

Salah satu tanda kemiskinan adalah keadaan rumah yang tidak layak huni. Dengan mengkaji masalah kesejahteraan masyarakat, Pemerintah Kota Tarakan menginisiasi kegiatan pemulihan tempat tinggal yang tidak layak huni bagi masyarakat kurang mampu melalui program New Singapore. Pada tahun 2012 dilakukan perbaikan 1000 unit rumah tidak layak huni.

Dalam APBN 2013, dana yang disiapkan untuk penyediaan rumah layak huni bagi masyarakat kurang mampu mencapai Rp 78 miliar. Program Rehabilitasi Rumah Tidak Layak Huni (RLTH), yang juga dikenal dengan renovasi rumah, merupakan salah satu komponen komitmen pemerintah dalam pengentasan kemiskinan melalui Dinas Sosial. Program renovasi rumah 2013 diluncurkan pada akhir Mei, dengan peletakan batu pertama.

Several features of livable aid have been researched. In the publication [3,] research data was gathered by directly visiting the Mesuji District Office and the poor who got aid to obtain data that corresponded to reality in the field through observations, interviews, and recording. Data is examined using an analytical descriptive technique after collection. Based on the findings of this research, the government-funded livable housing aid program in Mesuji Regency has been carried out correctly and in compliance with the activity's operational procedures. Meanwhile, according to Islamic economics, the root cause of poverty is a failure to meet the requirements of the community.

Ada beberapa metode untuk menentukan penerima manfaat rumah layak huni. Studi ini menggunakan pendekatan VIKOR, dan temuannya diharapkan dapat membantu pemerintah dalam mengelola Dana Bantuan RUTLAHU dan memilih pelamar yang memenuhi persyaratan. Berdasarkan temuan penelitian ini, pendekatan VIKOR dapat mengidentifikasi individu yang layak mendapatkan pembiayaan RUTLAHU dan melakukan pemeringkatan yang efektif [4].

Teknik weighted product (WP) telah diteliti untuk mengatasi tantangan terkait penyediaan bantuan rumah layak huni [5]. Data untuk penelitian ini dikumpulkan dari masyarakat Desa Pacinan per kepala keluarga. Selain itu, penulis mewawancarai seorang anggota staf kantor desa. Pada bulan Maret 2014, data dikumpulkan.. Dari hasil penelitian dapat didapatkan bahwa metode Weighted Product dapat diimplementasikan ke dalam sistem dan telah dibuktikan pada saat pengujian penelitian.

Metodologi TOPSIS, metode Sistem Pendukung Keputusan yang secara optimal/praktis dapat mendukung proses pengambilan keputusan dengan menggunakan ide-ide sederhana/mudah dipahami, digunakan dalam penelitian [6]. Data dari Desa Sumbaga digunakan dalam proses pemilihan bantuan rumah tidak layak huni, meliputi ciri-ciri dinding, atap, lantai, pekerjaan, pendapatan, jumlah tanggungan, luas rumah dan fasilitas MCK. Keberadaan SPK di Desa Sumbaga, menurut temuan studi, dapat membantu masyarakat menilai kelayakan mereka untuk mendapatkan bantuan RTLH dengan mempercepat dan mempersingkat berbagai proses.

Penelitian ini menggunakan teknik Fuzzy [1]. Dalam penelitian ini Fuzzy Multiple-Attribute Decision-Making (FMADM) digunakan untuk menemukan hasil dari pemilihan dan penghitungan setiap alternatif dengan Simple Additive Weighting (SAW). Pendekatan deskriptif atau survei digunakan dalam penelitian ini untuk mengumpulkan daftar nama yang diusulkan dari desa jamban sebagai data calon penerima bantuan rumah layak huni, serta berbagai kriteria yang akan dijadikan acuan dalam proses pengambilan keputusan. Berdasarkan temuan penelitian, tingkat akurasi metode SAW sebesar 95,44% dan tingkat akurasi metode FMADM sebesar 94,24%.

B. DATA MINING

Data Mining merupakan istilah untuk menggambarkan proses penemuan pengetahuan tersembunyi di dalam basis data. Data mining adalah suatu proses yang semi-otomatis untuk melakukan ekstraksi dan identifikasi informasi pengetahuan yang berpotensial dan berguna untuk disimpan dalam kumpulan data besar menggunakan pendekatan statistik, matematika, kecerdasan buatan, dan pembelajaran mesin [7]. Data mining, menurut Grup Gartner, adalah tindakan menemukan tautan, pola, dan tren yang signifikan dengan menganalisis sejumlah data dengan alat deteksi pola menggunakan perhitungan statistik dan matematis [8].

C. K-MEANS CLUSTERING

Salah satu algoritma clustering adalah K-Means Clustering. Clustering adalah teknik penambangan data yang membagi data menjadi banyak pengelompokan (grup, kluster, atau segmen), yang masing-masing dapat berisi beberapa anggota. Setiap objek ditugaskan ke grup yang paling mirip. Ini analog dengan mengelompokkan hewan dan tumbuhan ke dalam keluarga dengan anggota yang sebanding.

J.B. MacQueen mengembangkan algoritme K-Means pada tahun 1976, yang merupakan salah satu algoritme pengelompokan yang paling banyak digunakan untuk mengelompokkan data berdasarkan properti yang dapat

dibandingkan atau dibagi. Grup data ini dinamakan sebagai cluster. Data di dalam suatu cluster mempunyai ciri-ciri serupa dan tidak serupa dengan data cluster lain. Kompleksitas O(nKT) algoritma K-Means menunjukkan cukup efisien, dengan asumsi n adalah jumlah objek data, k adalah jumlah cluster yang dihasilkan, dan t adalah jumlah iterasi [9]. nilai k dan t biasanya jauh lebih kecil dari nilai n. Selanjutnya, algoritma ini akan berhenti pada situasi optimal selama iterasinya [10].

1) ALGORITMA K-MEANS CLUSTERING

Di bawah menggambarkan bagaimana algoritma K-Means membagi dataset menjadi cluster. [11]:

1. Tentukan k yang merupakan jumlah cluster yang dihasilkan. Untuk menghitung jumlah k cluster, perlu dilakukan pertimbangan teoretis dan kontekstual.
2. Secara acak menghasilkan k Centroid pertama (lokasi pusat cluster). Centroid awal dipilih secara acak dari antara item yang tersedia sebanyak k cluster, dan rumus berikut digunakan untuk menghasilkan centroid cluster ke-i berikutnya:

$$v = \frac{\sum_{i=1}^{n} x_i}{n}; i = 1,2,3,\dots n \quad (1)$$

Dimana ; v = centroid cluster
xi = objek yang ke-i
n = banyak objek yang menjadi anggota cluster

3. Tentukan jarak antara setiap objek dan setiap cluster centroid. Penulis menggunakan Euclidean Distance untuk menghitung jarak antara objek dan centroid.

$$d(x,y) = \|x-y\| = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}; i = 1,2,3,\dots n \quad (2)$$

Dimana ; xi = objek x yang ke-i
yi = daya y yang ke-i
n = jumlah objek

4. Setiap objek harus di alokasikan ke centroid terdekat. Lakukan iterasi, lalu temukan posisi centroid baru, untuk menetapkan objek ke setiap cluster selama iterasi.
5. Jika posisi centroid baru tidak sama dengan yang lama, ulangi langkah 3. Konvergensi diperiksa dengan membandingkan penentuan grup matriks pada iterasi sebelumnya dengan penentuan grup matriks pada iterasi saat ini. Jika hasilnya sama, maka algoritma k-means cluster analysis sudah konvergen; jika berbeda, proses belum konvergen, dan iterasi berikutnya diperlukan.

Sebelum proses analisa cluster, terlebih dahulu dilakukan perhitung mean dan standar deviasi untuk data yang sudah dikelompokkan dengan menggunakan persamaan 3 dan persamaan 4.

$$\mu = \frac{x_1+x_2+x_3+\dots+x_n}{n} \quad (3)$$

$$\sigma_{ki} = \sqrt{\frac{\sum_{j=1}^{n}(x_j-\mu_{ki})^2}{n-1}} \quad (4)$$

Dimana ; k = jumlah cluster 1…k
i = atribut 1 … i
n = banyaknya objek
xj = data ke-j
Ì = rata-rata cluster
μ = rata-rata cluster

Kepadatan klaster dapat digunakan untuk melakukan analisis klaster (cluster density). *Variance Within Cluster* (Vw) dan *Variance Between Cluster* (Vb) dapat digunakan untuk menghitung kepadatan cluster. Persamaan 5 digunakan untuk menghitung varian dari setiap pembentukan cluster.

$$Vc^2 = \frac{1}{n_c-1}\sum_{i=1}^{n}(y_i - \bar{y}_c)^2 \quad (5)$$

Dimana ; Vc = variance cluster c
c = 1..k, dengan k adalah banyak cluster
n_c = banyak data cluster c
y_i = data ke-I suatu cluster
y_c = rata-rata data suatu cluster

Dengan menggunakan rumus tersebut, kita dapat menentukan *variance within cluster* (Vw) dari varians di atas:

$$Vw = \frac{1}{N-c}\sum_{i=1}^{c}(n_i - 1){v_i}^2 \quad (6)$$

Dimana ; N = total data
c = 1..k, dengan k mewakili jumlah cluster
n_i = total data dalam suatu cluster
v_i = varian dalam suatu cluster

Lalu *Variance Between Cluster* (Vb) menggunakan persamaan:

$$Vb = \frac{1}{c-1}\sum_{i=1}^{c} n_i(\bar{y}_i - \bar{y})^2 \quad (7)$$

Dimana ; y = rata-rata dari y_i

Nilai variance digunakan untuk mengidentifikasi cluster ideal melalui perhitungan kerapatan cluster sebagai *variance within clusters* (Vw) dan *variance between clusters* (Vb) menggunakan persamaan 8.

$$V = \frac{Vw}{Vb} \quad (8)$$

D. LOGIKA FUZZY

Secara linguistik, *fuzzy* diartikan sebagai kabur atau. Nilai bisa sangat besar sementara juga salah. Dalam logika fuzzy, derajat memiliki rentang nilai dari 0 (nol) sampai 1 (satu).[12].

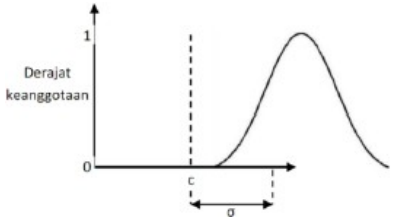
Logika fuzzy adalah penalaran dengan nilai antara benar dan salah. Akan tetapi, keberadaan dan kesalahan suatu ditentukan oleh besarnya keanggotaannya. Logika fuzzy memiliki derajat keanggotaan mulai dari 0 sampai

1) FUNGSI KEANGGOTAAN

ke dalam nilai penurunan (juga dikenal sebagai tingkat yang keanggotaan), yang memiliki rentang 0 hingga 1. Teknik fungsi adalah salah satu metode untuk menentukan nilai penarikan.

2) KURVA GAUSS

Fungsi keanggotaan Gaussian ditentukan dengan 2 parameter $\{c, \theta\}$ dengan mengikuti persamaan :



GAMBAR 1. **Alur Penelitian.**

Fungsi Keanggotaan:

$$G(x;k,\gamma) = e^{-k(\gamma-x)^2} \quad (9)$$

3) FUZZY INFERENCE SYSTEM (FIS) SUGENO

Metode fuzzy Sugeno adalah metode inferensi fuzzy untuk aturan yang didefinisikan sebagai *IF - THEN*, dimana output sistem (konsekuensi) adalah dalam bentuk konstanta atau persamaan linier daripada himpunan fuzzy. Takagi-Sugeno Kang memelopori pendekatan ini pada tahun 1985 [13]. Model Sugeno menggunakan fungsi fitur Singleton, yang memiliki derajat kehancuran satu untuk satu nilai crisp dan derajat kehancuran nol untuk nilai crisp lainnya. Formula Orde 0:

$$IF\ (x1\ is\ a1)°\ (x2\ is\ A2)°\ \dots °(xn\ is\ An)$$
$$THEN\ z =\ k, \quad (10)$$

Ai adalah himpunan fuzzy ke i sebagai antaseden (alasan). Sedangkan k adalah konstanta *assertive* sebagai konsekuen (kesimpulan), dan ° adalah operator fuzzy (AND atau OR). Sementara rumus Orde 1 adalah:

$$IF\ (x1\ is\ a1)°\ (x2\ is\ A2)°\ \dots °(xn\ is\ An)$$
$$THEN\ z\ =\ p1 * x1 + \cdots + pn * xn + q, \quad (11)$$

dimana Ai adalah himpunan fuzzy ke i sebagai antaseden, ° adalah operator fuzzy (AND atau OR), q merupakan konstanta dalam konsekuen, dan pi adalah konstanta ke I.

4) EKSTRAKSI ATURAN FUZZY

Aturan Fuzzy kadang-kadang dapat diperoleh dari ahli manusia. Akuisisi pengetahuan, bagaimanapun juga adalah tugas yang rumit, dan beberapa bagian sistem tidak diketahui, sistem pakar belum tersedia [14]. Data mining dapat didefinisikan sebagai proses otomatis mencari ukuran pola-pola data yang besar. Data mining menggunakan proses pencarian melalui ukuran data yang besar menggunakan teknik clustering (K-means, Fuzzy K-means, Subtractive) [15] untuk memperoleh data yang relevan dan signifikan dalam pengenalan pola, dan logika fuzzy dari fuzzy inferensi sistem (Mamdani dan TSK) merupakan teknik berbasis untuk mengesktraksi pembuatan aturan-aturan (IF- THEN) [16].

Proses pembangkitan aturan fuzzy dilakukan dengan proses pencarian nilai output melalui langkah-langkah berikut [17]:

- Menentukan derajat keanggotaan setiap titik data i dalam setiap cluster k dengan menggunakan fungsi gauss berdasarkan persamaan 12.

$$\mu ki = e^{-\sum_{j=1}^{m} \frac{(xij-Ckj)^2}{2\sigma_j^2}} \quad (12)$$

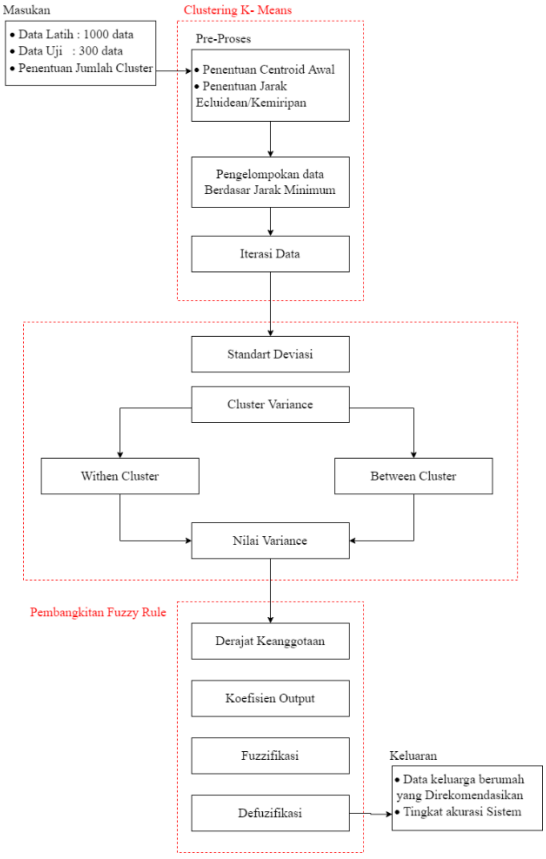
disusun menjadi satu vektor k:

$$k = [k_{11} \dots k_{1m} k_{10} k_{21} \dots k_{2m} k_{20} \dots k_{c1} \dots k_{cm} k_{10}]^T \quad (13)$$

Dari proses di atas terbentuklah koefisien output yang digunakan sebagai nilai output (konsekuen) dari inferensi Sugeno.

III. DESAIN ARSITEKTURAL

Arsitektur sistem digunakan untuk menggambarkan kerja sistem yang digunakan dalam proses analisis dan implementasi, serta arsitektur sistem secara keseluruhan yang terlihat pada Gambar 2.



GAMBAR 2. **Arsitektur Sistem**

1) MASUKAN

Masukan atau input merupakan data-data yang diperlukan selama proses analisa sampai implementasi sehingga bisa menghasilkan output yang sesuai dengan harapan. User memasukkan data keluarga untuk mendapatkan rekomendasi penerima bantuan rumah tidak layak huni dengan variabel seperti umur, penghasilan, jumlah tanggungan, kondisi lantai, kondisi dinding, kondisi atap dan kondisi WC, kemudian sistem melakukan proses clustering untuk mengelompokkan

data keluarga, untuk menghasilkan cluster yang ideal dilakukan perhitungan varian, within cluster dan between cluster.

2) CLUSTERING K-MEANS

Sebelum melakukan proses clustering, penulis menentukan terlebih dahulu centroid awal yang menjadi dasar clustering. Untuk melakukan pemilihan centroid awal caranya dengan menekan tombol cari centroid pada form iterasi, lalu tentukan centroid awal sacara random sesuai dengan jumlah cluster. Setelah melakukan pemilihan centroid awal, lalu lakukan perhitungan iterasi dengan menekan tombol "1" yang terdapat di group Hitung Iterasi.

3) ANALISA CLUSTER

Fitur analisis cluster digunakan untuk menghitung standar deviasi, varians, di dalam dan lintas cluster. Nilai varian cluster terkecil digunakan untuk menguji cluster ideal dengan menentukan nilai varian di dalam dan antar cluster.

4) PEMBANGKITAN FUZZY RULE

Fitur fuzzy digunakan untuk mengekstrak aturan fuzzy dengan menggunakan fuzzy inferece system sugeno. Hasil dari fuzzy inference system sugeno akan menghasilkan output rekomendasi penerima bantuan keluarga berumah tidak layak huni. Nilai jumlah cluster merupakan jumlah cluster yang digunakan untuk pengembangan sistem inferensi fuzzy aturan Sugeno, aturan yang dihasilkan jumlahnya sama dengan banyaknya cluster. Pada makalah ini, penulis menggunakan 3 (tiga) cluster, oleh karena itu aturan fuzzy akan dibentuk dengan 3 cluster (tiga) Jumlah data pelatihan yang digunakan, nilai varians, adalah nilai varians terkecil dari beberapa percobaan yang dilakukan penulis untuk membentuk cluster optimal. Fungsi tombol proses digunakan untuk melakukan proses mencari derajat keanggotaan kemudian langkah selanjutnya melakukan defuzifikasi.

5) KELUARAN

Output pada penelitian ini berupa rekomendasi penerima bantuan rumah tidak layak huni setelah dilakukan proses perhitungan dimana algoritma akan menghasilkan pengelompokkan data dengan kondisi yang optimum. Dalam kasus saran untuk penerima bantuan keluarga dengan perumahan tidak layak huni, K-means clustering digunakan untuk mengelompokkan data dan mengembangkan pedoman. Hasil dari pembangkitan aturan fuzzy digunakan untuk memproses inferensi fuzzy.

Sistem ini diharapkan memberikan sebuah hasil yang lebih relevan berupa rekomendasi penerima bantuan rumah tidak layak huni, sehinggan bisa dibandingkan dengan perhitungan yang dilakukan dinas social dan tenaga kerja di kota Tarakan.

IV. UJI COBA

Dalam proses uji coba yang dilakukan menggunakan data sampel 1000 data dan menggunakan 300 data uji. Dalam proses pengujian dilakukan dalam 20 kali uji sebanyak 300 data setiap skenarionya.

A. JUMLAH DATA UNTUK REKOMENDASI PEMBERIAN BANTUAN

Lokasi penelitian ini dilakukan pada Dinas Sosial dan Tenaga Kerja (Dinsosnaker). Alamat Dinas Sosisal dan Tenaga kerja Jl. Teuku Umar no.36 Kecematan Tarakan Tengah Kota Tarakan, Telp (0551) 21329-34499 Fax. 34499.

Sedangkan kurun waktu penelitian dilakukan sejak September tahun 2014 hingga Januari 2015. Pengambilan data input yang diperlukan dalam penelitian ini, diambil dari Dinsosnaker dengan mengambil data sejak tahun 2012, 2013 dan 2014.

B. JUMLAH DATA UNTUK REKOMENDASI PEMBERIAN BANTUAN

Pada penelitian ini dibutuhkan suatu sampel data yang nantinya diolah menjadi sebuah informasi. Sampel data ini merupakan sebagian dari populasi yang menjadi objek pengamatan dari penelitian ini.

Sampel data diambil dari bagian Penyandang Masalah Kesejahteraan Sosial (PMKS) rumah tidak layak huni yang ada di kota Tarakan. Untuk besaran bantuan yang diberikan mulai dari 13,5 juta dan 27 juta sesuai dengan kondisi rumah yang akan diberikan bantuan. Sampel data latih yang digunakan dalam proses k-means clustering sebanyak 1000 data yang diambil dari 4 Kecamatan yang ada di kota Tarakan yaitu Kecamatan Tarakan Barat yang terdiri antara delapan kecamatan, Desa Karang Anyar, Karang Anyar Pantai, Karang Harapan, Karang Balik, dan Karang Rejo. Kecamatan Tarakan Tengah dibagi lagi menjadi lima kecamatan: Desa Kampung Satu, Pamusian, Selumit, Selumit Pantai, dan Sebengkok. Kabupaten Tarakan Timur dibagi lagi menjadi tujuh kecamatan: Lingkas Ujung, Mambirdan, Kampung Empat, Enam Desa, Pantai Amal, Mamdindingan Timur, dan Gunung Lingkas. Kecamatan Tarakan Utara terbagi menjadi tiga kecamatan yaitu Juata Kerikil, Juata Laut, dan Juata Permai. Untuk data uji data yang digunakan sebanyak 300 data yang diambil dari 3 Kecamatan yang ada di kota Tarakan yaitu terdiri dari delapan kelurahan, yaitu Desa Karang Anyar, Karang Anyar Pantai, Karang Harapan, Karang Balik, dan Karang Rejo, disusul Kecamatan Tarakan Tengah yang terdiri dari lima desa, yaitu Kampung Satu, Pamusian, Selumit, Selumit Pantai, dan Desa Sebengkok, dan terakhir Kecamatan Tarakan Timur yang terdiri dari tiga kecamatan yaitu Kelurahan Gunung Lingkas, Mamburungan Timur dan Pantai Amal.

C. JUMLAH DATA UNTUK REKOMENDASI PEMBERIAN BANTUAN

Pada penelitian ini, data training yang digunakan sebanyak 1000 data dari data keluarga yang mengajukan permohonan untuk mendapatkan bantuan rumah tidak layak huni dari tahun 2012 dan tahun 2013 Data ini yang nantinya diproses untuk menghasilkan sebuah cluster untuk membentuk aturan fuzzy secara otomatis dengan menggunakan fuzzy inference system sugeno. Data uji yang digunakan adalah data keluarga

berumah tidak layak huni sebanyak 300 data dari keluarga yang mengajukan permohonan untuk mendapatkan bantuan rumah tidak layak huni tahun 2014.

D. PERHITUNGAN AKURASI

Hasil yang diamati pada penelitian ini adalah seberapa tingkat akurasi metode K-means clustering dalam membangkitkan aturan fuzzy dalam pengujian data bantuan keluarga berumah tidak layak huni. Dalam penelitian ini akurasi dihitung dengan cara membagi jumlah data uji dukungan keluarga terhadap rumah tidak layak huni dengan jumlah data. Akurasi merujuk pada seberapa dekat suatu hasil pengukuran dengan nilai sebenarnya (true value atau nilai referensi). Besarnya presisi dapat dihitung dengan menggunakan persamaan 11.

$$Akurasi = \frac{\sum Data\ Uji\ Benar}{\sum Total\ Data\ Uji} * 100 \quad (14)$$

E. HASIL PENGUJIAN

Dalam proses pengujian dilakukan pengujian sebanyak 300 data uji. Sedangkan untuk proses pembangkitan aturan fuzzy digunakan sebanyak 1000 data latih yang diambil dari data keluarga berumah tidak layak huni. Dan dilakukan perhitungan nilai varian untuk setiap jumlah cluster yang diuji. Nilai akurasi dilakukan dengan mencari rata-rata akurasi global untuk seluruh nilai akurasi skenario pada masing-masing jumlah cluster uji. Dari hasil proses pelatihan nantinya dicatat hasil perhitungan varian dan dibandingkan dengan nilai akurasi global untuk seluruh skenario di masing-masing jumlah cluster apakah nilai varian terkecil dari jumlah cluster memiliki nilai akurasi tertinggi.

TABEL I
HASIL PERHITUNGAN AKURASI

NO	ID	VARIANCE	JUMLAH CLUSTER	AKURASI
1	10,59,647	0,00111	3	82,33 %
2	16,59,622	0,00110	3	81 %
3	10,59,907	0,00126	3	81 %
4	382,300,948	0,00126	3	80,33 %
5	289,301,774	0,00126	3	79,33%
6	620,347,149	0,00126	3	79,33%
7	326,633,207	0,00126	3	79,33%
8	695,980,243	0,00127	3	78,33%
9	275,985,802	0,00127	3	78,33%
10	272,755,946	0,00127	3	78,33%
11	331,852,540	0,00127	3	78,33%
12	331,121,540	0,00127	3	78,33%
13	284,45,295	0,00127	3	78,33%
14	272,755,622	0,00127	3	78,33%
15	480,254,340	0,00128	3	77%
16	357,149,704	0,00129	3	76,33%
17	289,301,744	0,00129	3	76,33%
18	543,814,540	0,00131	3	74.33%
19	829,824,589	0,00131	3	74.33%
20	949,364,524	0,00131	3	74,33 %

Tabel 1 menampilkan hasil perhitungan varian yang digunakan untuk menentukan cluster ideal dengan menggunakan varian terkecil. Bisa dilihat untuk rata-rata akurasi global di setiap percobaan, menunjukkan bahwa pengujian akurasi paling maksimal dilakukan dengan menggunakan keluarga 10, keluarga 59 dan keluarga 647 dengan akurasi sitem 82 %.

V. KESIMPULAN

Dari penelitian berbasis K-Means clustering untuk pembangkitan aturan fuzzy dalam rangka rekomendasi pemberian bantuan rumah tidak layak huni yang telah dilakukan, dapat diambil kesimpulan:

1. K-Means clustering untuk pembangkitan aturan fuzzy merupakan metode yang tepat. Terbukti dari hasil perhitungan akurasi pengujian data uji setiap skenario dan akurasi pengujian data uji skenario global sama-sama menghasilkan akurasi minimal di atas 75%.
2. Dari hasil perhitungan nilai varian dan hasil perhitungan akurasi. Bisa diambil kesimpulan, proses penentuan jumlah cluster paling ideal masih belum optimal dengan menggunakan nilai varian terkecil. Dengan melihat hasil penelitian diperlukan adanya optimalisasi dalam menentukan cluster paling ideal.
3. Dalam prosedur ekstraksi aturan sugeno FIS, pendekatan pengelompokan K-Means dapat digunakan untuk membangun aturan fuzzy dengan penggunaan pusat cluster dan sigma.

Untuk pengembangan penelitian selanjutnya diperlukan adanya optimalisasi dalam menentukan cluster paling ideal. Selain itu, dapat pula dilakukan penyempurnaan proses pengelompokan yang bukan hanya pada satu periode saja, serta menambah kriteria-kriteria lain sehingga hasil yang diperoleh semakin sempurna.

DAFTAR PUSTAKA

[1] B. Satria and L. Tambunan, "Sistem Pendukung Keputusan Penerima Bantuan Rumah Layak Huni Menggunakan FMADM dan SAW," *JOINTECS (Journal of Information Technology and Computer Science)*, vol. 5, no. 3, pp. 167–176, 2020.

[2] "Wali Kota Tarakan Tinjau Program Bedah Rumah Agar Layak Huni - KANTOR BERITA KALIMANTAN." https://kbk.news/wali-kota-tarakan-tinjau-program-bedah-rumah-agar-layak-huni/ (accessed Nov. 17, 2022).

[3] K. Khotimah, "Analisis Program Bantuan Rumah Layak Huni Terhadap Pengentasan Kemiskinan di Kecamatan Mesuji Dalam Perspektif Ekonomi Islam," UIN Raden Intan Lampung, 2018.

[4] H. Tumanggor, M. Haloho, P. Ramadhani, and S. D. Nasution, "Penerapan Metode VIKOR Dalam Penentuan Penerima Dana Bantuan Rumah Tidak Layak Huni," *JURIKOM (Jurnal Riset Komputer)*, vol. 5, no. 1, pp. 71–78, 2018.

[5] D. Kusumawardani, "Sistem Pendukung Keputusan Penerima Bantuan Rumah Layak Huni Dengan Menggunakan Metode Weighted Product (WP)," *FASILKOM UDINUS*, 2014.

[6] H. Nalatissifa and Y. Ramdhani, "Sistem Penunjang Keputusan Menggunakan Metode Topsis Untuk Menentukan Kelayakan Bantuan Rumah Tidak Layak Huni (RTLH)," *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 19, no. 2, pp. 246–256, 2020.

[7] Efraim. Turban, J. E. Aronson, and T.-P. Liang, "Decision support systems and intelligent systems," p. 936, 2005.

[8] "Straubhaar, J. and LaRose, R. (2006) Communications Media in the Information Society. Wadsworth Publishing Company,

Belmont, CA. - References - Scientific Research Publishing." https://www.scirp.org/%28S%28351jmbntvnsjt1aadkozje%29%29/reference/referencespapers.aspx?referenceid=2515740 (accessed Nov. 18, 2022).
[9] K. A. A. Nazeer and M. P. Sebastian, "Improving the Accuracy and Efficiency of the k-means Clustering Algorithm," 2009.
[10] R. Arapoglou, K. Kolomvatsos, and S. Hadjiefthymiades, "Buyer agent decision process based on automatic fuzzy rules generation methods," *International Conference on Fuzzy Systems*, pp. 1–8, 2010.
[11] A. Bakri and M. F. M. Adini, "PENGELOMPOKAN DATA KAJI CUACA MENGGUNAKAN K-MEANS BAGI PERAMALAN TABURAN HUJAN.".
[12] K. Sri, "Aplikasi logika fuzzy untuk pendukung keputusan / Sri Kusumadewi, Hari Purnomo," 2010.
[13] J. Moreno, O. Castillo, J. Castro, L. Martinez, and P. Melin, "Data Mining for extraction of fuzzy IF-THEN rules using Mamdani and Takagi-Sugeno-Kang FIS," *Engineering Letters*, vol. 15, 2007.
[14] P. C. Chang and C.-H. Liu, "A TSK type fuzzy rule based system for stock price prediction," *Expert Syst. Appl.*, vol. 34, pp. 135–144, 2008.
[15] A. PRIYONO, M. Ridwan, A. ALIAS, R. Rahmat, A. Hassan, and M. Mohd Ali, "Generation of Fuzzy Rules with Subtractive Clustering," *Jurnal Teknologi*, vol. 43, p. 143, 2005, doi: 10.11113/jt.v43.782.
[16] E. Turban, E. McLean, and J. Wetherbe, "Information Technology for Management : Making Connections for Strategic Advantage / E. Turban, E. McLean, J. Wetherbe.," 2001.
[17] L. T. 1 Kóczy, "Fuzzy Rule Base Model Identification Techniques," 2005.

Metode Pembobotan Hibrida untuk Ekstraksi Frasa Kunci Bahasa Arab

Evan K. Susanto[1], M. Bahrul Subkhi[2], Agus Z. Arifin[2], Maryamah[2], Rizka W. Sholikah[3], dan Rarasmaya Indraswari[4]
[1]Departemen Informatika, Fakultas Sains dan Teknologi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Indonesia
[2]Departemen Teknik Informatika, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[3]Departemen Teknologi Informasi, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[4]Departemen Sistem Informasi, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

Corresponding author: Evan K. Susanto (e-mail: evanks@stts.edu).

ABSTRACT A large amount of information makes indexing and finding the intent of documents a challenging task. Most of the documents are also not associated with any unique keyphrases. Readers are therefore forced to read the entire document to get a whole picture of its contents. The automatic keyphrase extraction using YAKE Algorithm provides a fast solution of keyphrase extraction using a single document feature. However, using only local features results in the extraction results being less relevant because they require significant terms mentioned in other documents. Another problem that rises is that some local features can't be used for Arabic language, like capital letters. In this paper, we propose a hybrid term weighting method that integrates local statistical features from a single document and external documents for unsupervised arabic keyphrase extraction systems. This keyphrase extraction system can be effectively used in Arabic. and other languages that do not use capital letters and unstructured documents such as news or scientific papers. We show that our method performs better with experimental results than the baseline methods, which are YAKE and TF-IDF.

KEYWORDS Arabic Keyphrase Extraction, Hybrid Term Weighting Method, Information Retrieval, Unsupervised Algorithm.

ABSTRAK Banyaknya informasi membuat proses pengindeksan dan pencarian inti dari dokumen menjadi permasalahan yang rumit. Sebagian besar dokumen yang tersedia tidak dilengkapi dengan kata kunci terkait. Hal ini sehingga memaksa pembaca untuk membaca seluruh dokumen untuk mendapat gambaran penuh dari konten seluruh dokumen. Ekstraksi frasa kunci otomatis yang menggunakan Algoritma YAKE memberi solusi cepat ekstraksi frasa kunci menggunakan fitur lokal dari sebuah dokumen. Namun, penggunaan fitur lokal saja membuat hasil ekstraksi menjadi kurang relevan karena diperlukan istilah signifikan yang muncul di dokumen lain. Masalah lain yang muncul adalah terdapat beberapa fitur lokal yang tidak dapat digunakan untuk bahasa Arab, misalnya huruf kapital. Pada penelitian ini, diusulkan metode pembobotan kata yang mengintegrasikan fitur statistik lokal dari sebuah dokumen dan fitur eksternal dari dokumen lain untuk sistem ekstraksi kata kunci. Metode ini dapat digunakan secara efektif pada bahasa Arab dan dapat digunakan pada bahasa lain yang tidak memiliki huruf kapital serta untuk dokumen-dokumen yang tidak terstruktur seperti berita atau karya ilmiah. Dari hasil uji coba telah dibuktikan bahwa performansi metode ini lebih baik daripada metode pembanding yaitu YAKE dan TF-IDF.

KATA KUNCI Algoritma Unsupervised, Ekstraksi Frasa Kunci Bahasa Arab, Pembobotan Kata Metode Hibrida, Temu Kembali Informasi

I. PENDAHULUAN

Jumlah informasi yang banyak pada era digital ini membuat pengindeksan dan pencarian makna dari sebuah dokumen menjadi hal yang sulit dilakukan. Hampir semua dokumen tidak dilengkapi dengan frasa kunci sehingga pembaca perlu membaca seluruh isi dokumen agar dapat diperoleh informasi kunci yang ada di dalamnya. Proses pencarian frasa kunci ini memerlukan waktu yang sangat panjang dan usaha yang luar biasa apabila dilakukan secara manual. Jumlah dokumen yang semakin banyak menyebabkan ekstraksi manual frasa kunci dari sebuah dokumen menjadi tidak efisien.

Untuk mengatasi masalah tersebut, dikembangkanlah metode ekstraksi frasa kunci otomatis. Metode-metode ini biasanya dapat mencari 5 sampai 10 frasa kunci (yang masing-masing terdiri dari satu atau lebih kata) dari sebuah dokumen. Ekstraksi frasa kunci otomatis dapat digunakan untuk memberikan gambaran singkat tentang konten dari sebuah dokumen. Selain itu, frasa kunci ini juga sangat membantu untuk proses sistem temu kembali informasi.

Salah satu masalah yang dihadapi dalam proses ekstraksi frasa otomatis sebuah dokumen adalah pemrosesan bahasa yang digunakan dalam dokumen tersebut. Setiap bahasa memiliki karakteristik masing-masing. Tidak semua bahasa dapat dinormalisasi dengan mudah menjadi sebuah representasi yang universal untuk dapat diproses dengan menggunakan satu sistem yang sama.

Ekstraksi frasa kunci otomatis untuk dokumen dalam Bahasa Arab termasuk salah satu proses yang cukup menantang. Bahasa Arab digunakan oleh sebagian besar penduduk dunia sehingga jumlah publikasi dan dokumen yang menggunakan Bahasa Arab semakin meningkat dengan cepat hari demi hari. Namun, dataset dokumen Bahasa Arab terlabel yang tersedia masih sedikit. Bahasa Arab juga memiliki karakteristik unik seperti tidak adanya huruf kapital, penulisan dari kanan ke kiri, dan perlunya proses normalisasi khusus. Hal ini membuat proses normalisasi yang biasa dilakukan dalam dokumen lain tidak dapat diterapkan secara langsung pada dokumen berbahasa Arab.

II. *TEORI DASAR*

A. TF-IDF

TF-IDF [1] merupakan salah satu metode paling umum yang digunakan untuk mengambil konteks dari suatu teks [2] dengan cara merepresentasikan dokumen dalam bentuk angka. Sesuai dengan namanya, TF-IDF mendapatkan frasa dengan cara mengkalikan Term Frequency (TF) dengan Inverse Document Frequency (IDF). Frasa yang memiliki hasil nilai yang tinggi memiliki kemungkinan tinggi bahwa frasa tersebut dapat merepresentasikan dokumen tersebut, atau frasa tersebut merupakan frasa kunci.

Term Frequency (TF) menghitung berapa kali suatu frasa muncul pada dokumen tersebut. Cara ini dipakai karena pada umumnya seberapa sering suatu frasa muncul pada suatu kata berbanding lurus dengan relevansi frasa tersebut pada teks [3]. Nilai TF dapat didapatkan hanya dengan sekedar menghitung frekuensi suatu frasa, atau dapat menggunakan cara yang lebih kompleks [4] seperti melakukan normalisasi setelah mendapatkan frekuensi kemunculan frasa untuk memperhitungkan panjang dari suatu dokumen itu sendiri.

Inverse Document Frequency (IDF) adalah sebuah metrik yang menghitung invers dari frekuensi kemunculan sebuah kata pada sekumpulan dokumen[5]. Metrik ini digunakan untuk membuat kata-kata yang umumnya sering muncul pada teks, tetapi tidak memiliki relevansi pada teks untuk disaring keluar. Contoh pada kata Bahasa Inggris adalah kata-kata: "the", "and", dan "of". Kata-kata ini sering muncul di banyak dokumen sehingga kata-kata ini memiliki nilai IDF yang rendah. Sebaliknya, sebuah dokumen yang misalnya membahas tentang sebuah topik, misalnya “river”, akan banyak mengandung kata “river” tetapi kata “river” tidak banyak muncul di dokumen lain. Hal ini menyebabkan nilai IDF dari “river” akn menjadi tinggi. IDF membantu kata-kata relevan yang lebih jarang muncul dapat memiliki nilai TF-IDF lebih tinggi.

B. YAKE

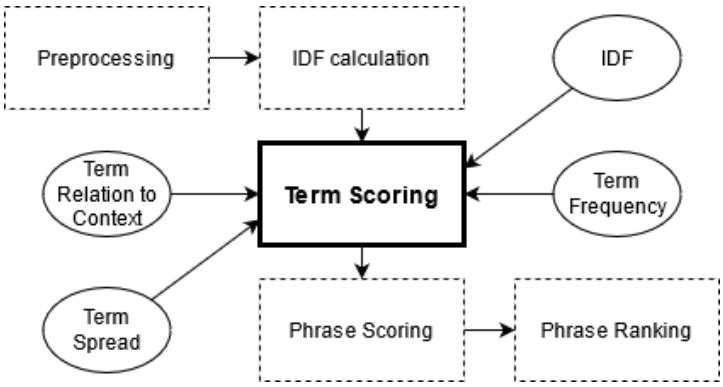
Berbeda dengan TF-IDF, YAKE [6] menghitung bobot untuk suatu kata menggunakan lima fitur dari dokumen dan perhitungan bobot yang dapat menggabungkan bobot dari beberapa kata untuk menghitung bobot suatu frasa. Kelima fitur ini merupakan: Term Casing, Term Position, Normalized Term Frequency, Term Relatedness to Context, dan Term Different Sentences. Persamaan untuk menghitung bobot dapat dilihat pada persamaan 1.

$$\mathbf{S}(\boldsymbol{t}) = \frac{\boldsymbol{T_{rel}} \times \boldsymbol{T_{position}}}{\boldsymbol{T_{case}} + \frac{\boldsymbol{TF_{norm}}}{\boldsymbol{T_{rel}}} + \frac{\boldsymbol{T_{sentence}}}{\boldsymbol{T_{rel}}}} \quad (1)$$

T_{rel} merupakan representasi seberapa dekat suatu kata dengan konteks teks tersebut. $T_{position}$ merupakan nilai posisi kata terhadap dokumen. T_{case} menghitung frekuensi kata tersebut muncul jika frasa tersebut muncul dengan huruf besar didepannya. TF_{norm} serupa dengan TF yang digunakan oleh TF-IDF, tetapi TF yang digunakan pada rumus adalah TF yang telah dinormalisasi. $T_{sentence}$ merupakan jumlah kemunculan relatif kata pada kalimat yang berbeda-beda.

T_{case} digunakan dikarenakan dalam beberapa bahasa, akronim atau kata-kata penting biasanya direpresentasikan dengan huruf besar atau huruf paling pertama dari kata tersebut merupakan huruf besar. Pada YAKE, T_{case} dihitung dengan cara menghitung berapa kali sebuah kata muncul diawali dengan huruf kapital atau semua kata-katanya terdiri dari huruf kapital. Semakin sering kata tersebut muncul dengan huruf besar semakin tinggi bobotnya[7].

$T_{position}$ mengasumsikan kata-kata yang muncul pada awal teks lebih relevan daripada jika kata-kata tersebut muncul



GAMBAR 1. **Diagram yang menggambarkan proses ekstraksi frasa kunci**

ditengah ataupun akhir teks [8]–[10]. Dikarenakan biasanya awal teks digunakan penulis untuk menarik perhatian pembaca, untuk menginformasikan kepada pembaca, tentang apa teks tersebut [3]. Selain itu, dokumen resmi seperti berita dan laporan ilmiah biasanya meletakkan kalimat utamanya di awal paragraf. Karena itu, tidak digunakan distribusi normal pada YAKE, tetapi menggunakan formula TF yang sudah dimodifikasi [4] untuk memperhitungkan bobot pada kalimat yang ada pada awal teks lebih besar daripada ditengah ataupun dibelakang teks [11].

Peran T_{rel} dalam YAKE adalah untuk menyaring stopwords didalam suatu teks. Untuk mengukur sebagaimana relevan suatu kata dalam teks tersebut. Apakah kata-kata yang sering muncul tersebut relevan terhadap teks [12] atau kata-kata yang lebih jarang muncul lebih relevan.

TF_{norm} adalah mengukur nilai frekuensi kemunculan kata dalam sebuah dokumen. Nilai ini mirip dengan nilai TF yang ada pada TF-IDF. Penggunaan metrik ini didasari dari asusmsi yang sama dengan TF-IDF bahwa sebuah kata yang muncul berulang kali pada sebuah dokumen (selain *stopwords*) kemungkinan besar merupakan frasa yang dapat mendeskripsikan dokumen tersebut [3]. Bedanya adalah pada rumus ini nilai TF dinormalisasi untuk mengurangi pengaruh dari panjang dokumen. Nilai dari TF_{norm} dihitung menggunakan rumus perhitungan frekuensi kata yang dimodifikasi [4], yaitu dari frekuensi kemunculan kata dibagi rata-rata kemunculan dari semua kata yang ada di dalam dokumen ditambah faktor normalisasi berupa satu standar deviasi.

$T_{sentence}$ menangani asumsi bahwa frasa kunci akan muncul dikalimat berbeda-beda didalam suatu teks. Asumsi ini didasari dari hal atau topik yang dibahas akan disebutkan berkali-kali pada teks dan merupakan frasa kunci. Nilai dari fitur ini dihitung dengan cara menghitung banyaknya kalimat yang mengandung kata ini dibagi dengan banyaknya kalimat yang ada pada dokumen

III. *METODOLOGI*

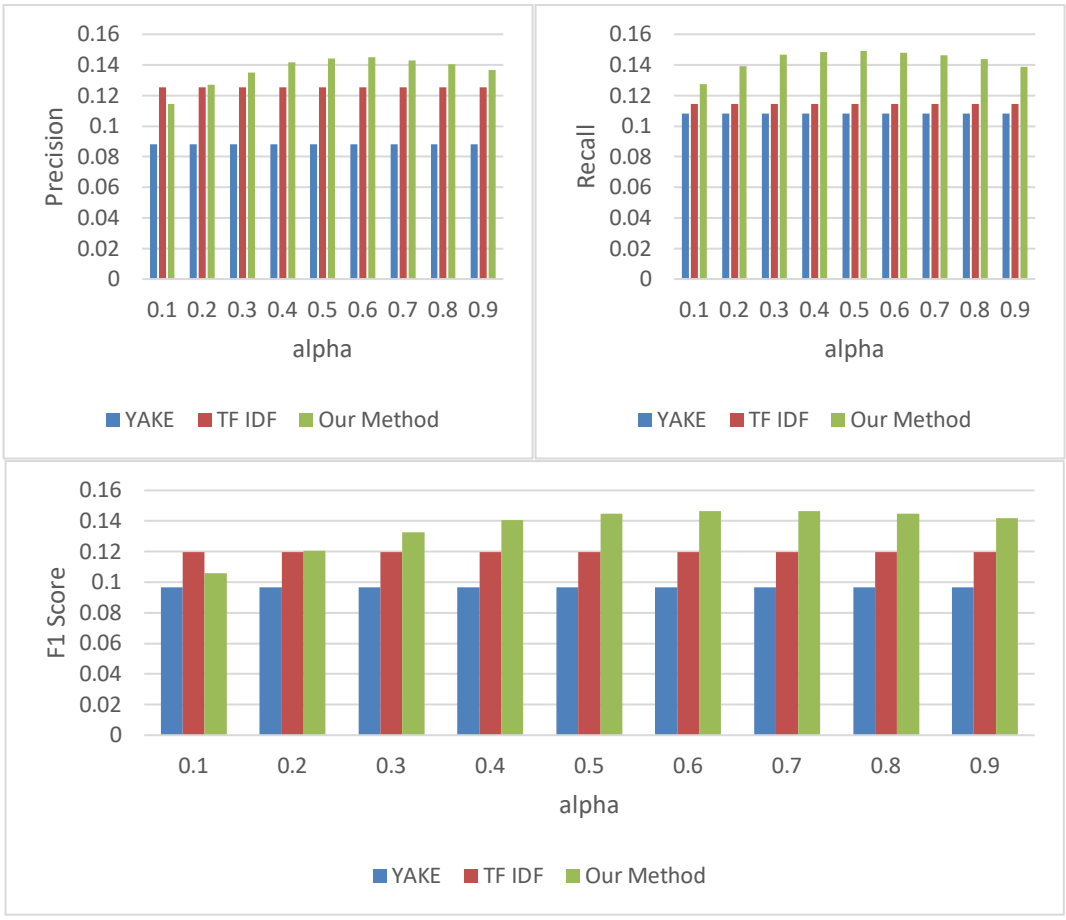
Pada bab ini, akan didiskusikan metode yang digunakan dalam penelitian ini. Akan dijelaskan langkah per langkah proses dari metode yang diajukan. Akan diperkenalkan juga metode pembobotan gabungan untuk menentukan sebagaimana mungkin suatu frasa merupakan frasa kunci dari suatu dokumen, yang didapatkan dari menggabungkan dua metode yang sudah dikenal yang menggunakan fitur lokal dan fitur eksternal dari sekumpulan dokumen. Ringkasan dari metode yang digunakan pada penelitian ini dapat dilihat pada Gambar 1.

Metode yang diajukan mengikuti alur dari algoritma YAKE, yang terdiri dari 5 langkah utama. Lima langkah utama tersebut adalah: preprocessing, term listng, term scoring, phrase scoring, dan phrase ranking. Kontribusi penelitian ini adalah melakukan modifikasi dalam langkah term scoring dengan cara melakukan integrasi pembobotan IDF yang didapat dari metode TF-IDF untuk meningkatkan kinerja pembobotan dalam YAKE. Di bab ini akan dijelaskan modifikasi yang dilakukan.

A. PREPROCESSING

Langkah pertama dalam penelitian ini adalah preprocessing. Pada langkah ini, dihapus semua angka dan tanda baca yang tidak menandakan akhir kalimat dari semua dokumen. Setelah itu, semua dokumen dipecah menjadi kalimat-kalimat. Pada penelitian ini tidak dilakukan preprocessing lain yang biasanya dilakukan pada algoritma lain seperti stemming atau melakukan normalisasi pada huruf Arab, karena hal ini dapat memberikan dampak negatif pada kualitas frasa kunci yang diekstraksi. Algoritma ini juga di rancang untuk dapat digunakan lintas bahasa, dapat digunakan walaupun hanya terdapat sedikit dokumen. Stemming atau normalisasi tidak digunakan karena mungkin saja tidak dapat diaplikasikan terdapat bahasa - bahasa lain.

Proses lain dalam preprocessing adalah mendaftar semua kata yang mungkin menjadi kata kunci atau bagian dari frasa kunci. Hal ini dilakukan dengan cara membuat set dari setiap kata (hanya satu per kata) yang muncul pada dataset. Pada langkah ini, dibutuhkan juga input yang terdiri dari kata-kata stopword yang digunakan pada bahasa target. Lalu setiap kata stopword yang muncul pada set kata-kata diberikan label. Setelah semua kata sudah didaftar dan semua stopword sudah diberikan label, proses akan dilanjutkan ke langkah perhitungan IDF.



GAMBAR 2. **Perbandingan performansi antara YAKE, TF-IDF, dan metode yang diusulkan menggunakan semua dokumen pada dataset**

B. KALKULASI IDF

Langkah berikutnya dari metode pada penelitian ini adalah menghitung nilai IDF untuk setiap kata-kata. Nilai IDF dihitung dari kata-kata menggunakan metode standar untuk menghitung IDF, yaitu menghitung log dari jumlah dokumen dibagi dengan jumlah dokumen yang mengandung kata-kata tersebut. Pertama, didaftar semua kata-kata pada dokumen. Lalu dihitung jumlah dokumen pada corpus, yang disimbolkan juga sebagai N. Terakhir, untuk setiap kata t, dihitung jumlah dokumen yang mengandung kata-kata t tersebut, yang disimbolkan sebagai n_t. Persamaan untuk perhitungan nilai IDF untuk suatu kata t (IDF_t) dapat dilihat pada Persamaan 2.

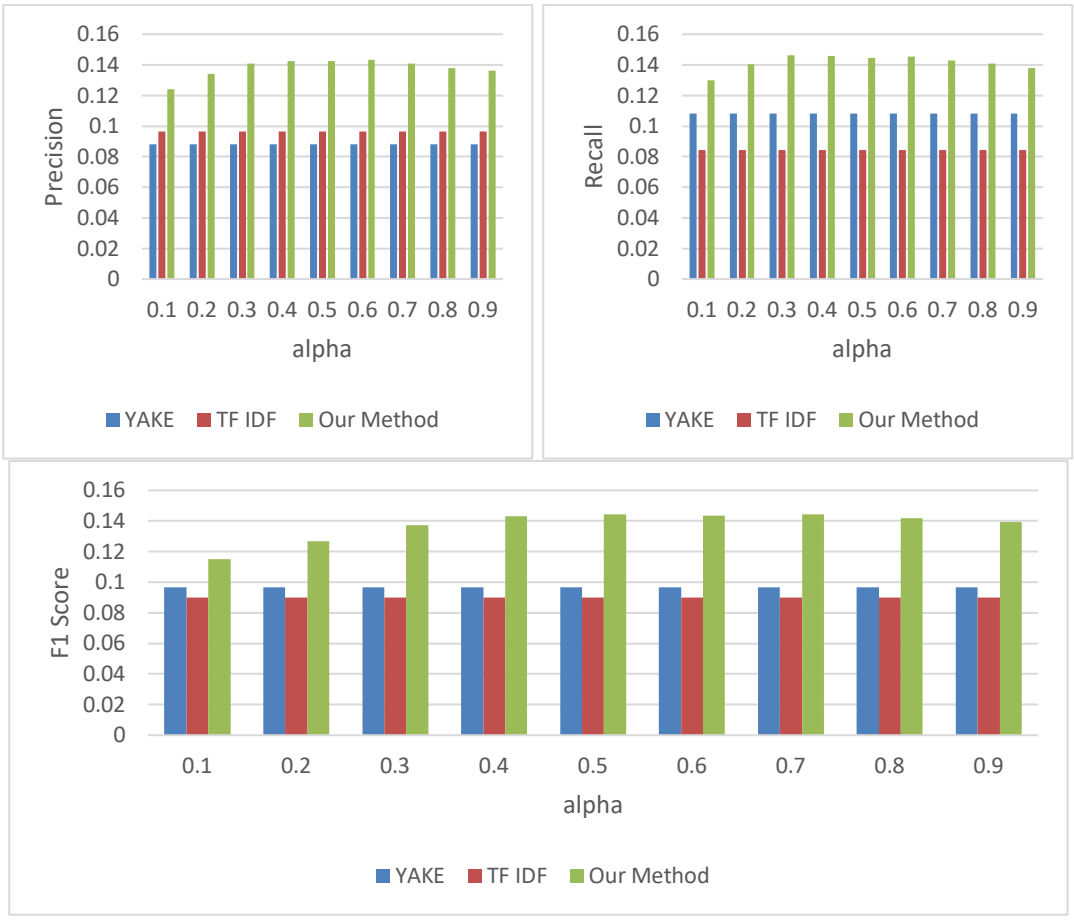
$$\boldsymbol{IDF_t = \log\left(\frac{N}{n_t}\right)} \quad (2)$$

C. TERM SCORING WITH HYBRID TERM WEIGHTING METHOD

Langkah ketiga dari proses ekstraksi frasa kunci pada penelitian ini adalah penilaian kata-kata. Pada penelitian ini diusulkan 3 modifikasi dari formula yang digunakan pada algoritma YAKE: (1) menghapus term casing dan term position dari formula YAKE, (2) menambahkan kalkulasi untuk menentukan seberapa penting suatu kata bedasarkan dokumen eksternal, dan (3) menambahkan hyperparameter yang dapat menyesuaikan pengaruh penilaian kata-kata terhadap dokumen target dan dokumen eksternal.

Modifikasi pertama adalah menghapuskan term casing dan term position dari formula YAKE. Term casing dihapus karena Bahasa Arab tidak memiliki huruf besar. Modifikasi ini tidak hanya berguna untuk Bahasa Arab, tetapi juga berguna untuk bahasa lain yang tidak memiliki huruf besar seperti Bahasa Mandarin, Bahasa Jepang, ataupun Bahasa Korea. Pada saat uji coba menggunakan YAKE, fitur ini menyebabkan YAKE selalu menganggap bahwa named entity merupakan frasa kunci, walaupun kenyataannya tidak demikian.

Term position juga dihapus karena term position mengasumsikan kata-kata pada awal dokumen lebih besar kemungkinannya kata-kata tersebut merupakan frasa kunci. Asumsi ini benar jika frasa kunci yang berusaha diekstrak dari dokumen formal seperti berita atau jurnal ilmiah. Tetapi, asumsi ini tidak benar jika pada dokumen yang tidak formal atau dokumen yang lebih tidak terstruktur seperti teks religius



GAMBAR 3. **Perbandingan performansi antara YAKE, TF-IDF, dan metode yang diusulkan menggunakan hanya sebagian dokumen pada dataset**

ataupun artikel di internet. Bobot ini dihapus untuk membuat pembobotan menjadi lebih umum dan dapat digunakan pada dokumen yang kurang terstruktur.

Modifikasi kedua memperkenalkan fitur baru yang merepresentasikan bobot dari suatu kata bedasarkan dokumen eksternal. Fitur ini dapat dianggap sebagai pengganti dari fitur casing karena keduanya dapat mendeteksi kata-kata penting. Banyak metode yang dapat digunakan untuk menghitung seberapa penting kata-kata bedasarkan dokumen eksternal, seperti Inverse Document Frequency (IDF), Inverse Class Frequency (ICF) [13], Inverse Book Frequency (IBF) [14], dan Inverse Preference Frequency (IPF) [15]. Dalam penelitian ini digunakan pembobotan IDF.

Modifikasi terakhir ada hyperparameter α yang dapat menyesuaikan pengaruh bobot yang dihitung dari dokumen target dan dokumen eksternal. Ide utama dari hyperparameter ini adalah, jika terdapat 2 algoritma berbeda untuk ekstraksi frasa kunci, satu menggunakan fitur dari dokumen target seperti YAKE, dan satu lagi menggunakan dokumen ekternal seperti TF-IDF, performa YAKE tetap sama tidak peduli seberapa banyak dokumen yang tersedia pada dataset. Tetapi sebaliknya, peforma TF-IDF membaik semakin banyak dokumen yang tersedia pada dataset. Tetapi peforma TF-IDF juga memburuk semakin sedikit dokumen yang tersedia pada dataset. Dengan menambahkan hyperparameter yang dapat mengatur pengaruh dari kedua cara ini bedasarkan jumlah dokumen pada dataset yang tersedia.

Secara total, terdapat 4 fitur yang diekstraksi untuk setiap kata untuk menghitung nilai dari kata tersebut, 4 fitur tersebut merupakan: IDF, frekuensi yang dinormalisasi, sebagaimana dekat kata tersebut dengan konteks, dan kemunculan kata dikalimat-kalimat yang berbeda. Nilai IDF telah dihitung pada langkah sebelumnya. Untuk tiga fitur berikutnya, akan dipinjam metode perhitungan algoritma YAKE. Frekuensi yang telah dinormalisasi didapatkan dari jumlah kemunculan kata didokumen dibagi dengan rata-rata jumlah kemunculan setiap kata ditambakan dengan satu standar deviasi.

$$W_{freq_t} = \frac{freq_t}{\overline{freq} + \sigma} \tag{3}$$

Untuk mendapatkan bobot ini untuk setiap kata, dikalkulasikan jumlah kemunculan tiap kata di dokumen tersebut. Setelah itu, dihitung rata-rata kemunculan kata dalam dokumen dan deviasi standarnya, yang disebut juga sebagai $\overline{freq}$ dan σ. Langkah terakhir, untuk setiap kata-kata t di dokumen, dihitung bobot frekuensi W_{freq_t}. Rumus yang

digunakan untuk menormalisasi term frequency dapat dilihat pada persamaan 3.

$$\boldsymbol{D} = \frac{|\boldsymbol{T_t}|}{\boldsymbol{\Sigma_{k \in T_t} Co(t,k)}} \quad (4)$$

$$\boldsymbol{W_{rel}} = \boldsymbol{1} + (\boldsymbol{D_L} + \boldsymbol{D_R}) \times \frac{\boldsymbol{freq_t}}{\boldsymbol{freq_{max}}} \quad (5)$$

D merepresentasikan faktor dispersi dari sebuah kata, T_t adalah set dari kata-kata yang muncul pada satu sisi (baik kanan maupun kiri), dan $Co(t,k)$ adalah jumlah kata t dan k muncul pada saat yang bersamaan. W_{rel} merepresentasikan seberapa mendekati konteks suatu kata, D_L dan D_R ada perhitungan faktor dispersi dari sisi kiri kata maupun sisi kanan kata, $freq_t$ adalah jumlah kemunculan kata t pada dokumen dan $freq_{max}$ adalah jumlah kemunculan maksimal dari suatu kata. Rumus perhitungan term relatedness to context dapat dilihat pada rumus 4 dan 5.

Perhitungan bobot terakhir yang digunakan pada kalkulasi ini ada spread weight dari suatu kata, yang dimana dihitung dengan cara membagi jumlah kalimat dalam dokumen yang mengandung kata target dengan jumlah kalimat dalam dokumen. Untuk menghitung spread weight (W_{spread_t}) dari kata t, pertama-tama dibuat set yang mengandung semua kalimat dalam dokumen yang disebut sebagai $sent$. Lalu dibuat set yang mengandung kata t, yang disebut juga sebagai $sent_t$. Berikutnya dihitung jumlah elemen pada $sent_t$ dan $sent$ untuk menghitung spread weight dari kata t (W_{spread_t}). Persamaan dari W_{spread_t}dapat dilihat pada rumus 6.

$$\boldsymbol{W_{spread_t}} = \frac{|\boldsymbol{sent_t}|}{|\boldsymbol{sent}|} \quad (6)$$

Keempat fitur ini digabungkan dengan hyperparameter α untuk membentuk rumus pembobotan yang dapat dilihat pada persamaan 7.

$$\boldsymbol{Score_t} = \frac{\boldsymbol{W_{rel}}}{\boldsymbol{\alpha(W_{freq}+W_{spread})+(1-\alpha)*W_{IDF}}} \quad (7)$$

$Score_t$ merepresentasikan nilai kata, W_{rel_t} merepresentasikan sebagaimana dekat kata dengan konteks, W_{freq_t} adalah bobot frekuensi yang telah di normalisasi, W_{spread_t} mereperesentasikan spread weight dari kata, W_{IDF_t} adalah nilai IDF dari kata, dan alpha adalah nilai hyperparameter penyesuaian dimana $0 < \alpha < 1$. Hyperparameter α dapat digunakan untuk mengatur fitur apa yang memiliki efek lebih tinggi terhadap metode penilaian. Semakin kecil α berarti fitur dari dokumen eksternal akan memiliki efek lebih tinggi, dan semakin tinggi α berarti fitur dokumen internal akan memiliki efek lebih tinggi terhadap pembobotan. Ini dikarenakan α akan dikalikan dengan $(W_{freq} + W_{spread})$, yang merupakan fitur dokumen internal dan $(1 - \alpha)$ akan dikalikan dengan W_{IDF}, yang merupakan fitur dokumen eksternal.

D. PHRASE SCORING

Setelah setiap bobot kata dalam dokumen dihitung, mulai dihitung nilai dari frasa kunci kandidat yang terdiri dari lebih dari satu kata. Dalam penelitian ini, frasa kunci yang dicari terdiri tidak lebih dari tiga kata tetapi bisa kurang dari tiga kata. Langkah pertama dalam proses ini adalah membuat daftar setiap frasa yang terdiri dari dua atau tiga kata yang muncul bersamaan. Lalu dihapus frasa yang dimulai atau diakhiri dengan stopword, dikarenakan frasa kunci jarang dimulai atau diakhiri dengan stopword [16]–[18].

Langkah sebelumnya akan menghasilkan tiga jenis frasa kunci: frasa yang terdiri dari tiga kata yang bukan merupakan stopword, frasa yang terdiri dari dua kata yang bukan merupakan stopword, dan frasa yang teridiri dari satu kata stopword diantara dua kata yang bukan merupakan stopword. Ketiga jenis frasa kunci ini kemudian digabung menjadi satu daftar frasa kunci dan menghitung nilai akhir setiap frasa kunci kandidat. Untuk frasa yang tidak mengandung stopword dapat dihitung nilai akhirnya menggunakan fumus 8.

$$\boldsymbol{Score_{kp}} = \frac{\boldsymbol{\Pi_{t \in kp}\, Score_t}}{\boldsymbol{freq_{kp} \times (1 + \Sigma_{t \in kp} Score_t)}} \quad (8)$$

Langkah pertama untuk menghitung nilai akhir adalah menghitung jumlah nilai $Score_t$ untuk setiap kata dalam frasa kunci kandidat kp. Lalu dihitung jumlah kemunculan frasa kunci kp pada dokumen ($freq_{kp}$). Terakhir, jumlah nilai dari setiap kata pada frasa kunci $\Pi_{t \in kp}\, Score_t$ dibagi dengan jumlah kemunculan frasa kunci $freq_{kp}$ dan 1 ditambahkan dengan jumlah nilai dari setiap kata pada frasa kunci $\Sigma_{t \in kp}\, Score_t$.

Digunakan rumus berbeda untuk menghidung kata kunci yang mengandung stopword ditengah. Nilai kata dari stopword diganti dengan dengan probabilitas kata pertama muncul sebelum stopword dan probabilitas kata terakhir muncul setelah stopword. Rumus yang digunakan untuk menghitung nilai frasa kunci kandidat yang mengandung stopword dapat dilihat pada rumus 9 dan 10.

$$\boldsymbol{S_{t_2}} = \boldsymbol{1 - P(t_2|t_1) \times P(t_3|t_2)} \quad (9)$$

$$\boldsymbol{Score_{kp}} = \frac{\boldsymbol{Score_{t_1} \times (1 + S_{t_2}) \times Score_{t_3}}}{\boldsymbol{freq_{kp} \times (1 + Score_{t_1} - S_{t_2} + Score_{t_3})}} \quad (10)$$

Pertama dihitung nilai dari stopword t_2 (S_{t_2}) dengan mengkalikan $P(t_2|t_1)$, probabilitas stopword t_2 muncul setelah kata pertama t_1, dengan $P(t_3|t_2)$, probabilitas kata

terakhir t_3 muncul setelah stopword t_2. Lalu digunakan rumus yang serupa dengan rumus 8 untuk menghitung nilai akhir, dengan beberapa perbedaan. Nilai $Score_{t_1}$ kata pertama dikalikan dengan nilai $Score_{t_2}$ kata kedua dan nilai $1 + S_{t_2}$ stopword. Kemudian hasil yang diperoleh dibagi dengan hasil perkalian dari jumlah kemunculan frasa kunci kandidat dengan hasil pertambahan nilai $Score_{t_1}$ kata pertama, nilai $Score_{t_3}$ kata terakhir dan 1, lalu mengurangi nya dengan nilai stopword S_{t_2}.

E. PHRASE RANKING

Langkah terakhir dari proses yang diusulkan pada penelitian ini adalah meranking setiap frasa kunci kandidat bedasarkan nilai mereka. Frasa kunci kandidat diurutkan bedasarkan nilai yang didapatkan dari 2 langkah sebelum ini dari kecil ke besar. Frasa kunci yang memiliki nilai lebih rendah akan memiliki ranking lebih tinggi dari yang memiliki nilai yang lebih tinggi. Lalu akan diambil frasa kunci tertinggi N sebagai frasa kunci.

Sebelum frasa kunci tertinggi N diambil, akan dilakukan penghapusan duplikat. Pada tahap ini terkadang masih ada frasa kunci yang mirip dengan frasa kunci lain pada daftar. Ini dikarenakan tidak dilakukannya proses stemming pada tahap preprocessing yang akan menggabungkan frasa kunci yang mirip menjadi satu. Digunakan metode Levenshtein distance untuk menghitung tingkat kemiripan dari dua frasa kunci yang ada di daftar. Jika kedua kata kunci memiliki tingkat kemiripan diatas nilai batas, frasa kunci yang memiliki nilai yang lebih tinggi (ranking lebih rendah) akan dihapus. Proses ini akan diulangi sampai tidak ada frasa kunci yang serupa pada frasa kunci kandidat tertinggi N. Setelah semua duplikasi telah dihapus, makan frasa kunci kandidat yang masuk dalam batas N akan menjadi frasa kunci yang berhasil di ekstraksi.

IV. HASIL EKSPERIMEN DAN DISKUSI

A. HASIL EKSPERIMEN

Terdapat 2 dataset yang digunakan untuk penelitian ini. Pertama adalah Arabic Keyphrase Extraction Corpus (AKEC) [19]. Dataset ini memiliki 160 dokumen dalam Bahasa arab dari 4 corpora dan dokumen-dokumen tersebut dapat dibagi menjadi 9 topik berbeda. Dataset kedua diambil dari [20]. Dataset ini memiliki 400 dokumen yang dapat dibagi menjadi 18 topik berbeda. Pada penelitian ini, kedua dataset ini digabung membentuk 1 dataset. Contoh data yang digunakan dapat dilihat pada Tabel 1.

Dilakukan perbandingan metode yang diajukan, YAKE, dan TF-IDF. Performa masing-masing metode dilihat dari menghitung penilaian F1 [21]. Nilai F1 didapatkan dengan cara menghitung akurasi rata-rata dan frasa yang dapat ditebak (recall). Persamaan untuk menghitung presisi, recall, dan F1 dapat dilihat di persamaan 11, 12, 13.

Agar dapat mendapatkan penilaian lebih akurat, preprocessing dilakukan juga kepada frasa kunci yang dibuat secara manual. Stopword, tanda baca, huruf yang berulang, dan angka dihapus dari frasa kunci yang dibuat secara manual. Lalu frasa kunci tersebut distemming menggunakan ISRI Arabic stemmer [22]. Selain itu frasa kunci yang diambil hanyalah frasa kunci yang terdiri dari maximal 3 kata.

TABEL I
CONTOH POTONGAN DOKUMEN DAN FRASA KUNCINYA

Potongan Dokumen	Frasa Kunci
جغرافيا ليسوتو. أبرز حقيقة جغرافية عن ليسوتو أنها الدولة الوحيدة المستقلة في العالم التي تقع كليًا فوق متر في الارتفاع. أخفض نقطة فيها ترتفع متر عن سطح البحر ما يجعلها أعلى أخفض نقطة في العالم لدولة ما. مناخ ليسوتو أكثر برودة من المناطق الأخرى في نفس خط العرض بسب ارتفاعها ويمكن تصنيفه في المناخات القارية.=الموقع=.ليسوتو بلد في أفريقيا الجنوبية ° وتقع عند خط عرض حوالي ° جنوبًا وبخط طول شرقًا...	'أعلى أخفض نقطة في العالم', 'جنوب أفريقيا', 'المناخ', 'التقسيم الإداري', 'جغرافيا ليسوتو', 'الأخطار البيئية', 'الموارد الطبيعية', 'دولة مغلقة', 'المناخات القارية'

$$\textbf{precision} = \frac{|\{\textbf{relevant kp}\} \cap \{\textbf{retrieved kp}\}|}{|\{\textbf{retrieved kp}\}|} \quad (11)$$

$$\textbf{recall} = \frac{|\{\textbf{relevant kp}\} \cap \{\textbf{retrieved kp}\}|}{|\{\textbf{relevant kp}\}|} \quad (12)$$

$$\textbf{F1 Score} = \frac{2 \times \textbf{precision} \times \textbf{recall}}{\textbf{precision} + \textbf{recall}} \quad (13)$$

Evaluasi dilakukan dengan menggunakan 2 skenario. Skenario pertama menggunakan semua dokumen training untuk menghitung IDF dan sebagai data testing. Skenario kedua, 25% dari total dataset digunakan untuk traning IDF dan 75% sisanya sebagai data testing. Skenario kedua digunakan untuk melakukan evaluasi jika hanya terdapat sedikit data yang tersedia untuk training IDF. Pada kedua skenario alpha yang digunakan adalah: 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9.

Pada Gambar 2 dapat dilihat algoritma yang diajukan bekerja lebih baik daripada YAKE dan TF-IDF pada hampir semua nilai alpha. Hal ini membuktikan dengan mengintegrasikan dokumen diluar dokumen target dapat meningkatkan kemampuan algoritma untuk ekstraksi frasa kunci dari suatu dokumen. Pada Gambar 3, dapat dilihat pada kasus dimana data training sedikit, performa TF-IDF memburuk, tetapi algoritma yang diajukan masih dapat bekerja lebih baik dari TF-IDF maupun YAKE. Hal ini

membuktikan dapat melakukan ekstraksi bergantung dari hanya dokumen target juga merupakan hal penting terutama pada kondisi dimana data training sedikit.

Dapat dilihat juga untuk setiap nilai α, performa algoritma yang diajukan lebih baik daripada YAKE. Maka dapat disimpulkan bahwa formula yang diajukan dapat melakukan ekstraksi lebih baik daripada YAKE untuk Bahasa Arab. Dikarenakan formula yang diajukan tidak lagi menggunakan T_{case} yang tidak relevan diakibatkan oleh sifat Bahasa Arab yang tidak memiliki kapitalisasi dan juga menggunakan IDF.

Dari kedua skenario dapat dilihat juga bahwa algoritma yang diajukan bekerja paling baik menggunakan α 0.5 dan 0.6. Hal ini disebabkan dengan nilai α demikian, maka akan dipertimbangkan dengan sama rata dokumen target dan dokumen lain. Penelitian lebih lanjut dapat dilakukan untuk memastikan apakah α yang optimal dapat berubah bedasarkan jumlah dokumen didalam corpus.

B. DISKUSI

Penelitian ini berusaha menggabungkan 2 metode penilaian untuk extraksi kata frasa yang bersifat unsupervised. Metode tersebut menggukan fitur lokal yang berasal dari dokumen input dan menggunakan fitur eksternal yang didapatkan dari kumpulan dokumen diluar dokumen input. Kedua metode mempunyai kelebihan yang membuat mereka populer, tetapi juga mempunyai kekurangan yang membuat mereka tidak cocok untuk diaplikasikan dalam kasus-kasus tertentu.

Algoritma yang menggunakan fitur lokal seperti YAKE dapat bekerja lebih baik dalam kasus dimana sangat sedikit atau bahkan tidak ada dokumen yang tersedia. Algoritma seperti ini biasanya universal, tidak terkunci disatu bahasa tertentu tanpa perlu modifikasi yang signifikan. Tetapi algoritma ini akan gagal pada kasus dimana sebagai mana penting frasa tidak dapat ditentukan hanya dari fitur lokal saja. Dikarenakan satu dokumen tidak dapat menyimpan semua informasi dan konteks dari suatu kata. Algoritma ini juga bisa gagal jika fitur yang diperlukan tidak ada (seperti pada kasus dimana YAKE tidak dapat ekstraksi penilaian yang didapat dari casing, dikarenakan Bahasa Arab tidak memiliki huruf besar).

Disisi lain, algoritma yang menggunakan informasi dari banyak dokumen dapat bekerja lebih baik dalam menentukan pembobotan kata dikarenakan algoritma tersebut dapat mendapatkan informasi dan konteks dari suatu kata lebih baik. Tetapi, algoritma ini akan gagal jika jumlah dokumen pendukung sedikit, dikarenakan algoritma ini tidak dapat mendapatkan cukup informasi mengenai suatu kata. Metode ini juga tidak dapat digunakan lintas bahasa dengan mudah seperti algoritma yang hanya menggunakan fitur lokal. Dilihat bahwa kedua algoritma ini memiliki kelebihan dan kekurangan yang dapat digabungkan untuk menutupi kekurangan yang lainnya.

Penelitian ini membuktikan dengan menggabungkan kedua metode ini dapat menghasilkan hasil yang lebih baik daripada kedua metode tersebut. Hal ini sangat menarik dikarenakan akan ada banyaknya kemungkinan area pengembangan yang dapat dieksplorasi lebih lanjut, seperti banyaknya jenis berbeda dari fitur lokal dan eksternal yang dapat digunakan untuk memodifikasi cara pembobotan. Posibilitas untuk mencari pengaturan terbaik pembobotan terbaik dari kedua jenis fitur juga dapat menjadi area penelitian yang menarik.

V. KESIMPULAN

Pada penelitian ini diusulkan sebuah metode pembobotan gabungan untuk ekstraksi frasa kunci. Pada penelitian ini diusulkan formula yang mengintegrasikan fitur statistik lokal yang didapatkan dari dokumen target dan fitur yang didapatkan dari dokumen pendukung lain. Pada penelitian ini juga diperkenalkan hyperparameter baru pada formula yang dapat mengatur pembobotan antara dokumen target dan dokumen pendukung untuk mengakomodasi jumlah dokumen pendukung yang digunakan untuk training. Pada penelitian ini juga ditunjukkan bahwa metode yang diusulkan memiliki peforma yang lebih baik daripada YAKE maupun TF-IDF. Pada penelitian ini juga ditunjukkan bahwa hyperparameter dapat meningkatkan peforma metode yang diusulkan bedasarkan jumlah dokumen pendukung. Pada penelitian ini juga dapat disimpulkan dari percobaan yang dilakukan bahwa nilai optimum untuk hyperparameter adalah diantara 0.5-0.6.

Terdapat beberapa pengembangan yang mungkin dapat diteliti kedepannya. Salah satunya adalah untuk mencoba metode pembobotan untuk fitur luar lain, seperti ICF [13], IBF [14], atau menggabungkan metrik-metrik lain [23]–[27]. Kemungkinan lain yang bisa dicoba adalah untuk mengaplikasikan metode yang diusulkan ini kepada bahasa lain atau jenis dokumen lain seperti artikel yang diambil dari internet atau teks religius. Untuk penelitian selanjutnya juga dapat mencoba untuk mendapatkan nilai hyperparameter yang optimal bedasarkan jumlah dokumen yang tersedia untuk training dan seberapa relevan dokumen training dan dokumen testing.

PERAN PENULIS

Evan Kusuma Susanto: Analisa permasalahan, investigasi, perumusan solusi, implementasi program, uji coba, penyusunan draf manuskrip,
M. Bahrul Subkhi: Analisa permasalahan, investigasi, perumusan solusi, penyusunan dataset, persiapan skenario uji coba, penyusunan draf manuskrip.
Agus Zainal Arifin, Maryamah, Rizka W. Sholikah, dan Rarasmaya Indraswari: Memiliki peran yang sama dalam membimbing investigasi, penyusunan solusi, dan penyusunan manuskrip.

COPYRIGHT

DAFTAR PUSTAKA

[1] G. Salton, "Automatic text processing: the transformation," *Anal. Retr. Inf. by Comput.*, 1989.
[2] J. Beel, B. Gipp, S. Langer, and C. Breitinger, "Research-paper recommender systems: a literature survey," *Int. J. Digit. Libr.*, 2016, doi: 10.1007/s00799-015-0156-0.
[3] H. P. Luhn, "A Statistical Approach to Mechanized Encoding and Searching of Literary Information," *IBM J. Res. Dev.*, 2010, doi: 10.1147/rd.14.0309.
[4] C. D. Manning, P. Raghavan, H. Schutze, C. D. Manning, P. Raghavan, and H. Schutze, "Scoring, term weighting, and the vector space model," in *Introduction to Information Retrieval*, 2012.
[5] K. Spärck Jones, "A statistical interpretation of term specificity and its application in retrieval," *J. Doc.*, 2004, doi: 10.1108/00220410410560573.
[6] R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes, and A. Jatowt, "YAKE! Keyword extraction from single documents using multiple local features," *Inf. Sci. (Ny).*, 2020, doi: 10.1016/j.ins.2019.09.013.
[7] R. Campos, V. Mangaravite, A. Pasquali, A. M. Jorge, C. Nunes, and A. Jatowt, "A text feature based automatic keyword extraction method for single documents," 2018, doi: 10.1007/978-3-319-76941-7_63.
[8] F. Boudin, "Unsupervised keyphrase extraction with multipartite graphs," 2018, doi: 10.18653/v1/n18-2105.
[9] A. Hulth, "Improved automatic keyword extraction given more linguistic knowledge," 2003, doi: 10.3115/1119355.1119383.
[10] C. Florescu and C. Caragea, "PositionRank: An unsupervised approach to keyphrase extraction from scholarly documents," 2017, doi: 10.18653/v1/P17-1102.
[11] R. Campos, V. Mangaravite, A. Pasquali, A. M. Jorge, C. Nunes, and A. Jatowt, "YAKE! collection-independent automatic keyword extractor," 2018, doi: 10.1007/978-3-319-76941-7_80.
[12] D. MacHado, T. Barbosa, S. Pais, B. Martins, and G. Dias, "Universal mobile information retrieval," 2009, doi: 10.1007/978-3-642-02710-9_38.
[13] F. Ren and M. G. Sohrab, "Class-indexing-based term weighting for automatic text classification," *Inf. Sci. (Ny).*, 2013, doi: 10.1016/j.ins.2013.02.029.
[14] M. A. Fauzi, A. Z. Arifin, and A. Yuniarti, "Arabic Book Retrieval using Class and Book Index Based Term Weighting," *Int. J. Electr. Comput. Eng.*, 2017, doi: 10.11591/ijece.v7i6.pp3705-3710.
[15] K. F. H. Holle, A. Z. Arifin, and D. Purwitasari, "Preference Based Term Weighting for Arabic Fiqh Document Ranking," *J. Ilmu Komput. dan Inf. (Journal Comput. Sci. Information)*, vol. 151, pp. 45–52, 2015, doi: http://dx.doi.org/10.21609/jiki.v8i1.283.
[16] S. Das Gollapalli, X. L. Li, and P. Yang, "Incorporating expert knowledge into keyphrase extraction," 2017.
[17] D. Mahata, J. Kuriakose, R. R. Shah, and R. Zimmermann, "Key2Vec: Automatic ranked keyphrase extraction from scientific articles using phrase embeddings," 2018, doi: 10.18653/v1/n18-2100.
[18] S. Rose, D. Engel, N. Cramer, and W. Cowley, "Automatic Keyword Extraction from Individual Documents," in *Text Mining: Applications and Theory*, 2010.
[19] M. Helmy, M. Basaldella, E. Maddalena, S. Mizzaro, and G. Demartini, "Towards building a standard dataset for Arabic keyphrase extraction evaluation," 2017, doi: 10.1109/IALP.2016.7875927.
[20] M. Al Logmani and H. Al Muhtaseb, "Arabic Dataset for Automatic Keyphrase Extraction," 2017, doi: 10.5121/csit.2017.70121.
[21] Y. Sasaki, "The truth of the F-measure," *Teach Tutor mater*, 2007.
[22] M. G. Syarief, O. T. Kurahman, A. F. Huda, and W. Darmalaksana, "Improving Arabic Stemmer: ISRI Stemmer," 2019, doi: 10.1109/ICWT47785.2019.8978248.
[23] E. Papagiannopoulou and G. Tsoumakas, "A review of keyphrase extraction," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 10, no. 2, p. e1339, 2020.
[24] Z. A. Merrouni, B. Frikh, and B. Ouhbi, "Automatic keyphrase extraction: a survey and trends," *J. Intell. Inf. Syst.*, vol. 54, no. 2, pp. 391–424, 2020.
[25] G. Rabby, S. Azad, M. Mahmud, K. Z. Zamli, and M. M. Rahman, "Teket: a tree-based unsupervised keyphrase extraction technique," *Cognit. Comput.*, vol. 12, no. 4, pp. 811–833, 2020.
[26] Y. Zhang, Y. Chang, X. Liu, S. Das Gollapalli, X. Li, and C. Xiao, "Mike: keyphrase extraction by integrating multidimensional information," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 1349–1358.
[27] E. Papagiannopoulou and G. Tsoumakas, "Local word vectors guiding keyphrase extraction," *Inf. Process. \& Manag.*, vol. 54, no. 6, pp. 888–902, 2018.

Klasifikasi Kategori Hasil Perhitungan Indeks Standar Pencemaran Udara dengan Gausian Naïve Bayes (Studi Kasus: ISPU DKI Jakarta 2020)

Devi Dwi Purwanto[1], **Eric Sugiharto Honggara**[1]
[1]Departemen Sistem Informasi, Fakultas Sains dan Teknologi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Indonesia

Corresponding author: Devi Dwi Purwanto (e-mail: devi@stts.edu).

ABSTRACT Air pollution is a problem that endangers humans, especially for the respiratory system. At present, air pollution always occurs due to several reasons such as vehicle, power plants and others. One of the places where air pollution occurs most is in big cities where many people gather. One of the places of concern is the station which is in the special area of the capital city of Jakarta. A station is a place where many people gather and wait to travel. Therefore, DKI Jakarta's environmental service open their data on air pollution that occurs at stations so that it can be used by the public for processing. The data will be preprocessed first by handling the missing values, then through data normalization and also used one hot encoding to uniform the data. The data will then be classified using the Gausian Naïve Bayes algorithm. After obtaining the results of the classification, it can be concluded that the max and critical attributes in the dataset have no effect on the clasification results for the ISPU category. The attributes of the data that influence the clasification of the ISPU category are PM10, SO_2, CO, O_3, and NO_2. By using 5 attributes and gausian naïve Bayes, the system can provide clasifications with an accuracy of 91.16% and an error rate of 8.84%. While the value of Weighted Average Recall 93,36%, Weighted Average Precision 93,92%, and Weighted Average F1-Score sebesar 93,68%.

KEYWORDS Gausian Naïve Bayes, Air Pollution, One Hot Encoding, Cross validation k-folds

ABSTRAK Pencemaran udara adalah masalah yang membahayakan manusia terutama untuk sistem pernafasan. Saat ini pencemaran udara selalu terjadi akibat beberapa hal seperti asap kendaraan, pembangkit listrik dan lainnya. Salah satu tempat di mana pencemaran udara terjadi adalah di kota besar di mana banyak orang berkumpul. Salah satu tempat yang menjadi perhatian adalah stasiun yang berada di daerah khusus ibukota jakarta. Stasiun adalah tempat di mana banyak orang berkumpul dan menunggu untuk melakukan perjalanan. Maka dari itu dinas lingkungan hidup DKI Jakarta membuka data pencemaran udara yang terjadi di stasiun agar dapat digunakan oleh masyarakat untuk diolah. Data tersebut akan dilakukan preprocessing yaitu penanganan missing value, normalisasi data, dan menggunakan one hot encoding. Data tersebut kemudian akan diklasifikasi dengan menggunakan algoritma Gausian Naïve Bayes. Setelah memperoleh hasil dari klasifikasi dapat disimpulkan bahwa atribut max dan critical yang berada dalam dataset tidak memiliki pengaruh terhadap hasil klasifikasi kategori ISPU. Atribut-atribut dari data yang berpengaruh terhadap klasifikasi kategori ISPU adalah PM10, SO_2, CO, O_3, dan NO_2. Dengan menggunakan 5 atribut dan gausian naïve bayes, sistem dapat memberikan klasifikasi dengan akurasi sebesar 91,16% dan memiliki error rate sebesar 8,84%. Sedangkan nilai Weighted Average Recall 93,36%, Weighted Average Precision 93,92% , dan Weighted Average F1-Score sebesar 93,68%.

KATA KUNCI Gausian Naïve Bayes, Polusi Udara, One Hot Encoding, Cross validation k-folds

I. PENDAHULUAN

Pencemaran udara adalah kehadiran sebuah subtansi fisik, kimia atau biologi dalam udara yang dapat menganggu kesehatan manusia. Secara spesifik, pencemaran ini dapat membuat manusia memiliki gangguan pernapasan, seperti asma, ISPA dan bahkan kanker paru-paru. Dinas lingkungan hidup Daerah Khusus Ibukota(DKI) Jakarta memiliki data yang telah dikumpulkan selama beberapa tahun dan ingin menggunakan data tersebut untuk melakukan klasifikasi kategori pencemaran udara yang terjadi pada ke-5 stasiun yang berada di kawasannya. Pencemaran udara yang terjadi di stasiun dikarenakan banyak faktor. Beberapa diantaranya adalah asap kendaraan bermotor [1], asap yang dihasilkan industri, pembangkit listrik dan rumah tangga, termasuk pencemaran udara karena debu yang berterbangan akibat aktifitas manusia yang berada di daerah sekitar stasiun. Maka pencemaran udara adalah sebuah masalah besar namun tidak kasat mata yang dapat membahayakan kesehatan, bahkan jiwa manusia dan hal ini tidak disadari oleh masyarakat yang tinggal dan menjalankan kehidupan mereka di kota besar [2]. Adapun pencemaran yang terjadi di dalam stasiun yang hari ini dapat terdeteksi adalah pecemaran PM10, SO_2, CO, O_3 dan NO_2. PM10, sebuah partikel di udara dengan ukuran lebih kecil dari 10 mikron dan dapat menyebabkan resiko karsinogenik [3]. SO_2 yang merupakan sulfur dioksida, sebuah senyawa kimia yang beracun bagi manusia dan biasanya berasa dari gunung atau hasil pemprosesan industri. SO_2 juga dapat menyebakan penurunan fungsi paru [4]. CO yang dikenal sebagai karbon monoksida, berupa gas. Gas CO ini tidak berwarna, tidak berbau, tidak berasa dan tidak memberikan rangsangan, oleh karena itu gas ini adalah gas yang susah dideteksi manusia dan berbahaya bagi kesehatan manusia [5]. O_3 adalah ozone, gas ini berbahaya bagi manusia karena dapat menyebabkan iritasi paru-paru dan tenggorokan, batuk dan memperburuk gejala asma [6]. Berikutnya adalah NO_2 atau Nitrogen Dioksida. Gas ini juga berbahaya bagi manusia karena dapat menyebabkan gangguan pernapasan seperti batuk, kemudian mata merah dan perih yang terjadi pada mata [7] [4]. Data yang dimiliki oleh dinas lingkungan hidup DKI Jakarta juga memiliki parameter max yang digunakan untuk melihat nilai ukur paling tinggi dalam waktu yang sama dan parameter critical yang digunakan untuk melihat mana parameter yang pengukurannya paling tinggi. Data ini belum dimanfaatkan sepenuhnya oleh dinas lingkungan hidup, oleh karena itu data tersebut dibuka kepada masyarakat agar bisa dimanfaatkan. Sehingga siapapun dapat melakukan pengolahan pada data yang ada agar dapat membantu Dinas Lingkungan Hidup DKI Jakarta untuk memprediksi pencemaran udara yang terjadi pada kelima statsiunnya.

II. LANDASAN TEORI

Berikut adalah landasan teori yang digunakan dalam melakukan pemprosesan data hingga memperoleh hasil sesuai tujuan.

A. ISPU

ISPU merupakan angka tanpa satuan yang digunakan untuk menggambarkan kondisi mutu udara ambien pada suatu lokasi yang didasarkan pada dampak terhadap kesehatan manusia dan makhluk hidup lainnya [8]. Menurut Peraturan Menteri Lingkungan Hidup dan Kehutanan no 45 tahun 1997 tentang Indeks Standar Pencemaran Udara ditetapkan bahwa ada 5 parameter yang mempengaruhi perhitungan ISPU yaitu PM10, NO2, SO2, CO, dan O3. Dari 5 parameter tersebut didapatkan nilai konversi parameter ISPU dan dikategorikan menjadi 5 kategori yaitu baik, sedang, tidak sehat, sangat tidak sehat, dan berbahaya. Tabel Kategori tersebut dapat dilihat pada tabel 1.

TABEL I
KATEGORI INDEKS STANDAR PENCEMARAN UDARA (ISPU)

RENTANG	KATEGORI	PENJELASAN
1-50	Baik	Tingkat mutu udara yang sangat baik, tidak memberikan efek negative terhadap manusia, hewan, dan tumbuhan.
51-100	Sedang	Tingkat mutu udara masih dapat diterima pada kesehatan manusia, hewan, dan tumbuhan.
101-200	Tidak Sehat	Tingkat mutu udara yang bersifat merugikan pada kesehatan manusia, hewan, dan tumbuhan.
201-300	Sangat Tidak Sehat	Tingkat mutu udara yang dapat meningkatkan resiko kesehatan pada sejumlah segmen populasi yang terpapar.
301+	Berbahaya	Tingkat mutu udara yang dapat merugikan kesehatan serius pada populasi dan perlu penanganan cepat.

Data pengukuran tersebut dilakukan selama 24 jam secara terus-menerus untuk mendapatkan 5 parameter tersebut dan nantinya ISPU yang diambil adalah perhitungan ISPU berdasarkan ISPU batas atas, batas bawah, ambien batas atas, ambien batas bawah, dan konsentrasi ambien hasil pengukuran. Persamaan perhitungan ISPU tersebut adalah sebagai berikut: [13]

$$I = \frac{I_a - I_b}{X_a - X_b}(X_x - X_b) + I_b \quad (1)$$

Dimana:
I = ISPU terhitung
I_a = ISPU batas atas
I_b = ISPU batas bawah
X_a = konsentrasi ambien batas atas (µg/m^3)
X_b = konsentrasi ambien batas bawah (µg/m^3)
X_x = konsentrasi ambien nyata hasil pengukuran (µg/m^3)

B. Data Preprocessing

Data preprocessing adalah proses mengubah data mentah ke dalam bentuk yang mudah dipahami dan siap untuk digunakan untuk proses berikutnya, karena data yang berkualitas akan berdampak pada keberhasilan terhadap proyek yang melibatkan analisa data. Proses dari Data

Preprocessing meliputi data cleaning, data integration, data transformation, dan data reduction. Proses ini juga disebut proses Extract, Transform, Load atau ETL yang juga dilakukan pada Data Warehouse. Fungsi utama dari ETL ini adalah mengurangi waktu reposn dan meningkatkan performa [9]. Tujuan akhir dari datawarehouse adalah menyiapkan sebuah data yang siap diproses.

Data cleaning adalah proses untuk membersihkan data yang missing value, menghaluskan data yang tidak pada umumnya, dan menyelesaikan data yang tidak konsisten yang terdapat di dalam dataset. Untuk penanganan missing value dapat dilakukan dengan beberapa cara yaitu menghilangkan data tersebut, mengganti dengan variable tertentu, mengisi dengan rata-rata atribut tersebut, mengisi dengan rata-rata atribut pada kelas yang sama, ataupun melakukan regresi untuk mengganti isi data yang kosong.

Data integration adalah tahap yang menggabungkan data dari berbagai sumber menjadi satu kesatuan data, dimana perlu diperhatikan bahwa saat digabungkan data harus memiliki format yang sama, melakukan deteksi nilai data yang konflik, dan menghapus atribut yang tidak dibutuhkan pada proses selanjutnya. Penghapusan atribut tersebut didasarkan pada tujuan melakukan mining.

Data transformation adalah proses untuk melakukan normalisasi dan generalisasi untuk memastikan bahwa tidak ada data yang berada di luar range dan menyeragamkan range data agar tidak terjadi ketimpangan. Hal ini dikarenakan data yang memiliki range yang timpang akan menyebabkan impact yang berbeda, dimana semakin besar valuenya akan semakin besar impact atribut tersebut.

Data reduction adalah proses pengurangan jumlah data, pengurangan dimensi, ataupun melakukan kompresi data sehingga data yang akan digunakan nantinya tidak menyebabkan akurasi menjadi rendah.

C. *One Hot Encoding*

One hot encoding adalah sebuah proses di mana variabel-variabel yang ada diubah menjadi sebuah bentuk yang bisa digunakan algoritma *machine learning* untuk melakukan klasifikasi dengan lebih baik. Bentuk dari hasil encoding ini adalah 1 dan 0 [10]. One hot encoding digunakan untuk data yang tidak memiliki relasi satu sama lainnya dan keunggulan utamanya adalah kemudahan dalam melakukan skala data. Untuk data dari dinas lingkungan hidup, data yang diubah dengan one hot encoding adalah parameter stasiun yang isinya hanya terdiri dari salah satu dari 5 stasiun di DKI Jakarta yakni "Stasiun_DKI1 (Bundaran HI)", "Stasiun_DKI2 (Kelapa Gading)", "Stasiun_DKI3 (Jagakarsa)", "Stasiun_DKI4 (Lubang Buaya)", dan "Stasiun_DKI5 (Kebon Jeruk)". Data tersebut akan diubah menjadi bentuk biner 1 untuk kategori yang muncul dan nilai 0 untuk kategori lain.

S_DKI1	S_DKI2	S_DKI3	S_DKI4	S_DKI5
1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1

GAMBAR 1. Contoh Perubahan One Hot Encoding

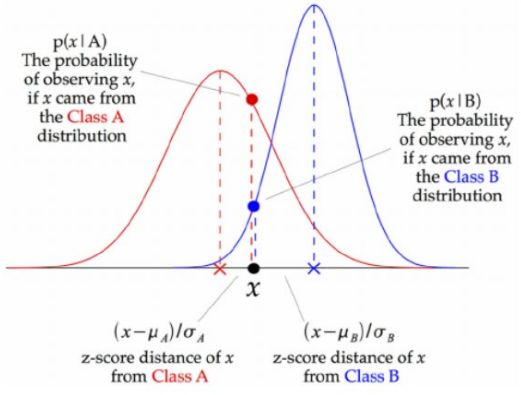
D. *Gausian Naïve Bayes*

Naïve Bayes adalah algoritma sederhana yang memanfaatkan aturan yang ada dalam bayes. Algoritma ini digunakan dengan asumsi bahwa atribut-atribut yang ada dalam data yang dimiliki tidak bergantung satu dengan yang lain. Algoritma Naïve Bayes memiliki peforma yang sangat baik untuk melakukan klasifikasi terutama dalam akurasi dari klasifikasi yang diberikan **[11] [12]**.

Gausian Naïve Bayes merupakan salah satu dari varian Naïve Bayes yang mengikuti distribusi normal gausian dan digunakan untuk data kontinu dengan rumus:

$$\boldsymbol{P(y) = \frac{1}{\sqrt{2\pi\sigma^2 y}} exp^{\left(-\frac{(x_i-\mu_y)^2}{2\sigma^2 y}\right)}} \quad (2)$$

Pendekatan untuk membuat model sederhana adalah dengan mengasumsikan bahwa data dideskripsikan oleh distribusi Gaussian tanpa co-variance antar dimensi. Dengan menggunakan model ini dapat dicocokkan hanya dengan menemukan rata-rata dan standar deviasi dari titik-titik dalam setiap label.



GAMBAR 2. Ilustrasi Klasifikasi dengan Gausian Naïve Bayes

Ilustrasi pada gambar 2 menunjukkan cara kerja pengklasifikasi Gaussian Naive Bayes (GNB) di mana pada setiap titik data dihitung jarak z-score antara titik tersebut dengan rata-rata kelas, yaitu jarak dari rata-rata kelas dibagi dengan standar deviasi kelas tersebut. Sehingga dapat dikatakan Gaussian Naive Bayes memiliki pendekatan yang sedikit berbeda dan dapat digunakan secara efisien pada data kontinu. Langkah selanjutnya untuk menentukan klasifikasi kelas akan dilakukan perhitungan probabilitas untuk masing-

masing atribut dengan memperhatikan teorema bayes, dimana untuk mendapatkan pengetahuan baru digunakan peluang statistika dari masing-masing atribut. Jika diasumsikan suatu kelas dengan C, maka dapat diketahui Probabilitas Class Prior P(C_i) dengan C_i adalah label kelas ke-i dan i = 1, 2, ..., m, sehingga menjadi:

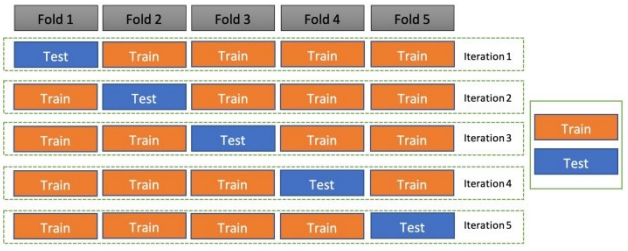
$$P(C_i) = \frac{N_c}{N} \quad (3)$$

Setelah mendapatkan probabilitas dari masing-masing kelas tersebut dan probabilitas class prior akan dilakukan perhitungan probabilitas akhir antara data testing dengan kelas target, hasil probabilitas yang akan diambil adalah probabilitas yang terbesar dan value dari kelas tersebutlah yang akan digunakan untuk klasifikasi kelas targetnya.

$$Pred = argmax_{C_i \in C} \frac{P(C_i)x\ P(C_i)}{P(x_1, x_{2,\ldots,} x_n)} \quad (4)$$

E. Cross Validation

Cross validation merupakan salah satu metode evaluasi untuk membandingkan antara data asli dengan data hasil klasifikasi dengan membagi menjadi 2 segmen yaitu data training dan data testing dan dilakukan secara acak. Metode ini merupakan salah satu metode yang biasanya digunakan jika terdapat jumlah value kelas target dari dataset yang inbalance. Metode ini dikenal dengan sebutan k-fold cross validation, dimana k adalah banyak iterasi untuk membagi data training dan testing. Ilustrasi k-fold cross validation dapat dilihat pada gambar 3.



GAMBAR 3. K-fold cross validation

III. Hasil

Pada bagian ini akan dijelaskan mengenakan tahapan-tahapan yang akan dilakukan untuk dapat melakukan klasifikasi pada dataset yang digunakan sebagai studi kasus. Tahapan tersebut adalah pengumpulan data, preprocessing, Data Mining, dan Evaluasi.

A. *Pengumpulan Data*

Dataset yang digunakan pada penelitian ini adalah open dataset dari Dinas Lingkungan Hidup Provinsi DKI Jakarta pada tahun 2020, dan diterbitkan dengan frekuensi penerbitan 1 bulan sekali yang diambil dari web https://data.jakarta.go.id/dataset/indeks-standar-pencemaran-udara-ispu-tahun-2020. Dataset ini mengambil sampel 5 stasiun besar yang ada di DKI Jakarta dengan atribut yang dicatat adalah tanggal pengambilan data, nama stasiun pengambilan data, partikulat salah satu parameter yang diukur(pm10), Sulfida dalam bentuk SO_2, carbon monoksida, ozon (O_3), nitrogen (NO_2), Nilai ukur paling tinggi dari seluruh parameter yang diukur dalam waktu yang sama(Max), Parameter yang hasil pengukurannya paling tinggi (critical), dan kategori. Dataset ini nantinya akan dibedakan menjadi data training dan data testing, dimana data training diambil dari ISPU bulan Januari hingga Agustus 2020 namun data pada bulan ke-3 tidak digunakan. Hal ini dikarenakan pada bulan ke-3 data yang diperoleh tidak ada nama stasiun tempat data tersebut dikumpulkan, sehingga data tidak dapat digunakan untuk melakukan klasifikasi pencemaran udara pada stasiun tersebut. Pada dataset tersebut hanya menggunakan 3 value dari 5 value kategori ISPU yang ditetapkan oleh Dinas Lingkungan Hidup dan Kehutanan. Tiga value tersebut yaitu baik, sedang, dan tidak sehat.

B. *Preprocessing*

Data preprocessing merupakan langkah awal yang perlu dilakukan dalam proses data mining agar data yang akan digunakan nantinya tidak bernoise, lengkap, dan formatnya terstruktur. Atribut yang akan digunakan nantinya adalah stasiun, PM10, NO2, SO2, CO, dan O3. Data preprocessing yang akan dilakukan pada bagian ini meliputi penanganan missing value pada semua atribut yang digunakan, normalization, dan one hot encoding.

GAMBAR 4. Preprocessing Raw Data

Sebelum data diproses maka data akan dilakukan pemeriksaan untuk meningkatkan akurasi. Dataset yang digunakan memiliki sebuah kekurangan, yaitu dataset tersebut terdapat missing value pada gambar 5 yang ditandai dengan '---'.

```
2020-08-13,DKI4 (Lubang Buaya),81,24,16,37,4,81,PM10,SEDANG
2020-08-14,DKI4 (Lubang Buaya),---,15,14,36,8,36,O3,BAIK
2020-08-15,DKI4 (Lubang Buaya),---,23,21,35,9,35,O3,BAIK
2020-08-16,DKI4 (Lubang Buaya),---,24,11,54,8,54,O3,SEDANG
2020-08-17,DKI4 (Lubang Buaya),---,22,17,50,7,50,O3,BAIK
```

GAMBAR 5. Bentuk Data Pencemaran Udara Dalam Stasiun

Penanganan missing value untuk 1 atribut yang kosong ini akan diganti dengan menghitung nilai rata-rata dari value yang ada dalam dataset untuk menggantikan missing value tersebut. Selain missing value, masalah kedua yang ada dalam dataset adalah adanya hari di mana data tersebut tidak ada. Untuk penanganan data yang tidak ada sama sekali dalam satu hari ini akan dilakukan penghapusan terhadap data tersebut. Sehingga dataset dapat diproses tanpa ada kendala.

```
2020-08-01,DKI4 (Lubang Buaya),---,---,---,---,---,0,,TIDAK ADA DATA
```

GAMBAR 6. Bentuk Data dengan Missing Value

Dapat dilihat pada gambar 6, karena seluruh data pada hari tersebut berupa garis maka data tersebut akan dihapus. Total data training yang digunakan ada sebanyak 1215 data setelah dilakukan pembersihan dan menghapus data yang "Tidak ada data" maka jumlah total akhir data sebanyak 1206 data, dimana detail perbandingan data training dapat dilihat pada gambar 7.

stasiun	categori	
DKI1 (Bunderan HI)	BAIK	81
	SEDANG	159
	TIDAK SEHAT	3
DKI2 (Kelapa Gading)	BAIK	45
	SEDANG	177
	TIDAK SEHAT	19
DKI3 (Jagakarsa)	BAIK	36
	SEDANG	189
	TIDAK SEHAT	18
DKI4 (Lubang Buaya)	BAIK	28
	SEDANG	198
	TIDAK SEHAT	10
DKI5 (Kebon Jeruk) Jakarta Barat	BAIK	21
	SEDANG	181
	TIDAK SEHAT	41

GAMBAR 7. Detail Perbandingan Data Training

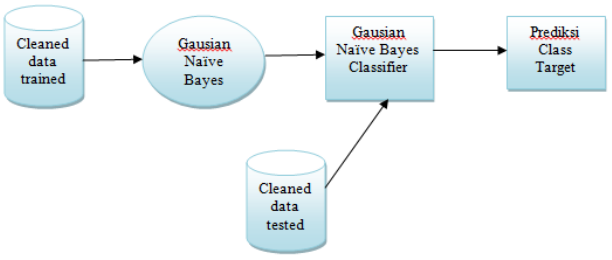
Langkah berikutnya adalah melakukan normalisasi data pada range yang sama. Setiap nilai dalam dataset memiliki value yang bervariatif dan memiliki batas atas dan bawah yang berbeda. Sehingga perlu dilakukan normalisasi agar data tersebut berada pada range yang sama. Untuk data PM10, SO_2, CO, NO_2, O_3 dan max akan dilakukan normalisasi dengan menggunakan min-max normalization dengan range nilai yang baru setelah dilakukan normalisasi yaitu 0-1. Rumus min max normalization yang digunakan adalah sebagai berikut

$$v' = \frac{v - (A)}{(A) - (A)}(new\_min\ (A) - new\_min\ (A)) + new\_min\ (A) \quad (5)$$

Sedangkan pada nama stasiun akan dilakukan one hot encoding sehingga data tersebut berubah menjadi bentuk yang bisa diolah dengan lebih mudah dengan menggunakan Gausian Naïve Bayes.

C. *Klasifikasi*

Tahapan berikutnya setelah melakukan preprocessing adalah klasifikasi. Tahapan klasifikasi tersebut dapat dilihat pada gambar 8. Dimana setelah melakukan data preprocessing dihasilkan cleaned Data, dimana cleaned data tersebut kemudian diproses dengan Gausian Naïve Bayes dan akan menghasilkan Gausian Naïve Bayes Classifier. Gausian Naïve Bayes Classifier tersebut nantinya akan digunakan untuk melakukan klasifikasi terhadap data testing yang diberikan dengan memperhatikan stasiun dan 5 parameterISPU.



GAMBAR 8. Tahapan Klasifikasi

Gausian Naïve Bayes Classifier yang dihasilkan disini berupa rata-rata tiap kelas, dan standar deviasi tiap kelas. Hal ini dikarenakan semua atribut yang digunakan berupa data kontinu, hanya 1 atribut yang bukan data kontinu yaitu atribut stasiun yang akan langsung dihitung probabilitasnya setelah dilakukan one hot encoding. Dari classifier tersebut nantinya digunakan untuk melakukan klasifikasi pada data testing yang diberikan. Dari data testing yang diberikan tersebut akan dihitung distribusi normal gausian untuk masing-masing atribut terhadap kelas untu setiap data testing. Setelah itu akan dilakukan perhitungan probabilitasnya untuk menentukan klasifikasi dengan menggunakan Gausian Naïve Bayes tersebut. Nilai probabilitas yang tertinggi tersebut yang akan diambil dan digunakan sebagai klasifikasi kelasnya. Hasil klasifikasi dengan menggunakan Gausian Naïve Bayes tersebut dapat dilihat pada gambar 9.

	stasiun	pm10	so2	co	o3	no2	max	critical	categori	prediksi
DKI1	(Bunderan HI)	45	18	4	49	10	68	PM25	SEDANG	SEDANG
DKI1	(Bunderan HI)	52	19	4	42	10	71	PM25	SEDANG	SEDANG
DKI1	(Bunderan HI)	60	22	5	68	11	86	PM25	SEDANG	SEDANG
DKI1	(Bunderan HI)	44	19	3	35	6	56	PM25	SEDANG	SEDANG
DKI1	(Bunderan HI)	46	18	8	64	9	64	O3	SEDANG	SEDANG
DKI1	(Bunderan HI)	37	15	6	82	11	82	O3	SEDANG	SEDANG
DKI1	(Bunderan HI)	54	20	7	78	12	78	O3	SEDANG	SEDANG
DKI1	(Bunderan HI)	57	17	6	82	14	82	O3	SEDANG	SEDANG
DKI1	(Bunderan HI)	62	19	6	65	11	91	PM25	SEDANG	SEDANG
DKI1	(Bunderan HI)	69	19	10	91	14	103	PM25	TIDAK SEHAT	TIDAK SEHAT
DKI1	(Bunderan HI)	61	17	11	77	13	87	PM25	SEDANG	SEDANG
DKI1	(Bunderan HI)	42	16	5	70	10	70	O3	SEDANG	SEDANG
DKI1	(Bunderan HI)	52	19	6	66	11	72	PM25	SEDANG	SEDANG
DKI1	(Bunderan HI)	55	19	6	47	11	76	PM25	SEDANG	SEDANG
DKI2	(Kelapa Gading)	69	25	8	70	13	96	PM25	SEDANG	SEDANG
DKI2	(Kelapa Gading)	72	21	10	122	11	122	O3	TIDAK SEHAT	TIDAK SEHAT
DKI2	(Kelapa Gading)	80	21	12	94	12	94	O3	SEDANG	SEDANG
DKI2	(Kelapa Gading)	56	21	6	51	9	73	PM25	SEDANG	SEDANG
DKI2	(Kelapa Gading)	54	22	6	118	7	118	O3	TIDAK SEHAT	TIDAK SEHAT
DKI2	(Kelapa Gading)	67	21	12	121	11	121	O3	TIDAK SEHAT	TIDAK SEHAT
DKI2	(Kelapa Gading)	69	26	8	84	13	101	PM25	TIDAK SEHAT	SEDANG

GAMBAR 9. Contoh Hasil Klasifikasi dengan Gausian Naïve Bayes

IV. Uji Coba

Setelah melakukan klasifikasi, proses berikutnya adalah pembentukan summary hasil klasifikasi terhadap data aktual yang mana nantinya digunakan untuk menghitung akurasi dari hasil klasifikasi terhadap data testing yang diujicobakan. Untuk mengatasi inbalanced data akan digunakan cross validation dengan k-fold=10. Hasil evaluasi model dapat dilihat pada gambar 10. Dari situ dapat dihitung pula akurasi yang diperoleh dengan menggunakan Gaussian Naïve Bayes

pada data lingkungan hidup DKI Jakarta untuk 5 stasiun yaitu sebesar 93,8%.

```
akurasi k-0=0.8760330578512396
akurasi k-1=0.9008264462809917
akurasi k-2=0.9421487603305785
akurasi k-3=0.9256198347107438
akurasi k-4=0.9173553719008265
akurasi k-5=0.9752066115702479
akurasi k-6=0.975
akurasi k-7=0.9416666666666667
akurasi k-8=0.95
akurasi k-9=0.975


Cross-Validation accuracy: 0.938
```

GAMBAR 10. Hasil Evaluasi dan Confusion Matrix

Selain melakukan klasifikasi, juga dilakukan uji coba terhadap atribut mana yang berpengaruh dan tidak terhadap penetuan kategori ISPU DKI Jakarta. Dimana data mentah yang diperoleh dari dataset memiliki 9 atribut dan 1 kelas target. Namun dalam datase tersebut terdapat beberapa atribut yang tidak bisa digunakan pada penelitian ini yaitu atribut max, critical, dan tanggal. Dimana pada pengujian jika algoritma Gaussian Naïve Bayes menggunakan atribut max dan critical hasil klasifikasi menjadi kurang akurat dan tingkat akurasi yang dihasilkan menurun hingga 66,7%. Sedangkan tanggal tidak memberikan pengaruh apapun pada tingkat akurasi. Confusion matrix dan akurasi tersebut dapat dilihat pada gambar 11.

```
akurasi k-0=0.5950413223140496
akurasi k-1=0.7107438016528925
akurasi k-2=0.4297520661157025
akurasi k-3=0.8760330578512396
akurasi k-4=0.7107438016528925
akurasi k-5=0.5950413223140496
akurasi k-6=0.725
akurasi k-7=0.6583333333333333
akurasi k-8=0.7
akurasi k-9=0.7666666666666667


Cross-Validation accuracy: 0.677
```

GAMBAR 11. Hasil Evaluasi dan Confusion Matrix dengan semua atribut

Dari hasil uji coba dengan menggunakan k-fold=10 untuk 5 atribut dilakukan perhitungan Recall, Precision, dan F1 Score. Hasil Recall, Precision dan F1 score dapat dilihat pada Tabel II. Didapatkan hasil WA Recall, Precision, dan F1 Score dengan 5 atribut sebesar 91%, sedangkan dengan menggunakan 7 atribut didapatkan WA Recall sebesar 65%, WA Precision 77%, dan WA F1 Score 62%.

TABEL II
PERHITUNGAN RECALL, PRECISION, DAN F1 SCORE

Iterasi		Baik	Sedang	Tidak Sehat	Weighted Average
1	Recall	0,86	0,89	1	0,876
	Precision	0,90	0,84	1	0,876
	F1 Score	0,88	0,86	1	0,874
2	Recall	0,88	0,96	0,6	0,901
	Precision	0,97	0,83	1	0,913
	F1-Score	0,92	0,89	0,75	0,9
3	Recall	0,88	0,94	0,90	0,928
	Precision	0,78	0,97	0,82	0,931
	F1-Score	0,82	0,95	0,86	0,925
4	Recall	1	0,95	0,75	0,924
	Precision	1	0,94	0,79	0,924
	F1-Score	1	0,95	0,77	0,927
5	Recall	0,65	0,97	0,92	0,92
	Precision	1	0,93	0,8	0,926
	F1-Score	0,79	0,95	0,86	0,918
6	Recall	0,92	0,99	1	0,976
	Precision	0,96	0,98	1	0,976
	F1-Score	0,94	0,98	1	0,972
7	Recall	0,91	0,98	1	0,974
	Precision	0,83	0,99	1	0,976
	F1-Score	0,87	0,99	1	0,979
8	Recall	1	0,98	0,55	0,941
	Precision	0,75	0,95	0,86	0,937
	F1-Score	0,86	0,97	0,67	0,94
9	Recall	0,83	0,99	0,71	0,949
	Precision	0,83	0,95	1	0,95
	F1-Score	0,83	0,97	0,83	0,947
10	Recall	1	0,99	0,71	0,982
	Precision	0,83	0,98	1	0,983
	F1-Score	0,91	0,99	0,83	0,986
Weighted Average	Recall	0,9336			
	Precision	0,9392			
	F1-Score	0,9368			

V. KESIMPULAN

Dari hasil uji coba yang telah dilakukan dalam melakukan klasifikasi kategori ISPU dengan menggunakan Gausian Naïve Bayes pada data lingkungan hidup 5 stasiun di DKI Jakarta dapat disimpukan:

1. Atribut max dan critical yang berada dalam dataset tidak memiliki pengaruh terhadap hasil klasifikasi kategori ISPU, terbukti dengan akurasi yang didapatkan bila mengikutkan semua atribut adalah 67,7%.
2. Atribut-atribut dari data yang berpengaruh terhadap klasifikasi kategori ISPU adalah PM10, SO2, CO, O3, dan NO2.
3. Dengan menggunakan 5 atribut dan gausian naïve bayes, sistem dapat memberikan klasifikasi dengan akurasi sebesar 93,8% dan memiliki error rate sebesar 6,2%. Sedangkan nilai WA Recall 93,36%, WA Precision 93,92% , dan WA F1 Score sebesar 93,68%.

PERAN PENULIS

Devi Dwi Purwanto: Konseptualisasi, metodologi, perangkat lunak, validasi, investigasi, kurasi data, penyusunan draft asli.

Eric Sugiharto: Investigasi, validasi, Analisis Formal, Investigasi, peninjauan dan penyuntingan

COPYRIGHT

DAFTAR PUSTAKA

[1] Indrayani and S. Asfiati, "Pencemaran Udara Akibat Kinerja Lalulintas Kendaraan Bermotor di Kota Medan," *Jurnal Pemukiman,* pp. 108-112, 2018.

[2] I. Ma'rufi, "Analisis Risiko Kesehatan Lingkungan(SO2 , H2S, NO2, dan TSP) Akibat Transportasi Kendaraan Bermotor di Kota Surabaya," *Media Pharmaceutica Indonesiana,* pp. 189-196, 2017.

[3] R. A. Lestari, A. R. Handik and S. I. Purwaningrum, "Analisis Risiko Karsinogenik Paparan PM10 Terhadap Pedagang di Kelurahan Pasar Jambi," *Jurnal Dampak,* pp. 59-65, 2019.

[4] A. Masito, "ANALISIS RISIKO KUALITAS UDARA AMBIEN (NO2 DAN SO2) DAN GANGGUAN PERNAPASAN PADA MASYARAKAT DI WILAYAH KALIANAK SURABAYA," *Jurnal Kesehatan Lingkungan,* pp. 394-401, 2018.

[5] L. M. Saleh, Keselamatan dan Kesehatan Kerja Kelautan : (Kajian Keselamatan dan Kesehatan Kerja Sektor Maritim), Deepublish Publisher, 2018.

[6] D. Nuvolone , D. Petri and F. Voller, "The Effects of Ozone on Human Health," *Enviromental Science and Pollution Research,* pp. 8074-8088, 2017.

[7] R. Darmawan, "ANALISIS RISIKO KESEHATAN LINGKUNGAN KADAR NO2 SERTA KELUHAN KESEHATAN PETUGAS PEMUNGUT KARCIS TOL," *Jurnal Kesehatan Lingkungan,* pp. 116-126, 2018.

[8] J. Abidin and F. A. Hasibuan, "Pengaruh Dampak Pencemaran Udara Terhadap Kesehatan untuk Menambah Pemahaman Masyarakat Awam Tentang Bahaya Dari Polusi Udara," in *Prosiding Seminar Nasional Fisika Universitas Riau IV*, Pekanbaru, 2019.

[9] V. Goar, S. S. Sarangdevot, G. Tanwar and A. Sharma, "Improve Performance of Extract, Transform and Load(ETL) in Data Warehouse," *International Journal on Computer Science and Engineering,* pp. 786-789, 2010.

[10] D. Harris and S. Harris, Digital Design and Computer Architecture, San Francisco: Morgan Kaufmann, 2012.

[11] M. M. Saritas and A. Yasar, "Perfromance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification," *International Journal of Intelligent Systems and Applications in Engineering,* pp. 88-91, 2019.

[12] A. A. Mahran, R. K. Hapsari and H. Nugroho, "Penerapan Naive Bayes Gaussian Pada Klasifikasi Jenis Jamur Bedasarkan Ciri Statistik Orde Pertama," *Networking Engineering Research Operation,* pp. 91-99, 2020.

[13] Kementerian Lingkungan Hidup dan Kehutanan, "WEB PORTAL DIREKTORAT PENGENDALIAN PENCEMARANUDARA," [Online]. Available: https://ditppu.menlhk.go.id/portal/read/indeks-standar-pencemar-udara-ispu-sebagai-informasi-mutu-udara-ambien-di-indonesia. [Accessed 20 December 2022].

Penyaring Komentar Cyberbullying Pada Konten Blog

Danar Dono [1], Eka Rahayu Setyaningsih [1], dan C. Pickerling [1]
[1]Departemen Teknologi Informasi, Fakultas Sains dan Teknologi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Indonesia

Corresponding author: Eka Rahayu Setyaningsih (e-mail: eka@stts.edu).

ABSTRACT Cyberbullying is a real threat to the interaction between blog content writers and blog readers. This research discusses the development of a cyberbullying filter feature on blog content to minimize cyberbullying on a blog site. The system development method uses an iterative waterfall including system analysis, system design, implementation, and testing. Based on testing with data training mode using 7755 comment datasets with a proportion of 3984 cyberbullying and 3771 non-cyberbullying results in an accuracy of 85.25% and an error of 14.75%. Testing with data testing mode using 1936 dataset comments with the proportion of 583 cyberbullying and 1353 non-cyberbullying resulted in 80% accuracy and 20% error. Based on the test, it can be concluded that the development of the cyberbullying comment filter feature using the Naive Bayes classifier produces an average accuracy of 80% and an average error of 20%.

KEYWORDS *Cyberbullying*, Comment Filtering, Classifier

ABSTRAK *Cyberbullying* merupakan ancaman nyata dalam interaksi di antara penulis konten blog dan pembaca blog. Penelitian ini membahas tentang pengembangan fitur penyaring *cyberbullying* pada konten blog untuk meminimalisir *cyberbullying* dalam suatu situs blog. Adapun metode pengembangan sistem menggunakan *iterative waterfall* meliputi analisis sistem, desain sistem, implementasi dan pengujian. Berdasarkan pengujian dengan mode pelatihan data menggunakan 7755 dataset komentar dengan proporsi 3984 *cyberbullying* 3771 non-*cyberbullying* menghasilkan akurasi 85,25% dan error 14,75%. Pengujian dengan mode testing data menggunakan 1936 dataset komentar dengan proporsi 583 *cyberbullying* dan 1353 non-*cyberbullying* menghasilkan akurasi 80% dan error 20%. Dari hasil pengujian disimpulkan bahwa pengembangan fitur penyaring komentar *cyberbullying* dengan menggunakan naive bayes classifier menghasilkan rata-rata akurasi sebesar 80% dan rata-rata error sebesar 20%.

KATA KUNCI *Cyberbullying*, Penyaringan Komentar, Classifier

I. PENDAHULUAN

Dewasa ini ada banyak sistem manajemen konten (CMS) yang digunakan untuk mengelola konten suatu situs web. CMS biasanya dilengkapi dengan fitur komentar bagi penggunanya, memungkinkan terjadinya interaksi antara pembuat konten dan pembaca artikel [1]. Komentar dapat berisi cyberbullying dalam konten blog. Cyberbullying adalah perlakuan kejam yang disengaja kepada orang lain dengan mengirimkan atau mengedarkan materi berbahaya atau terlibat dalam bentuk agresi sosial menggunakan internet atau teknologi digital lainnya [2]. Cyberbullying berdampak negatif pada korban trauma psikologis, emosional dan sosial. Fitur filter komentar cyberbullying diperlukan untuk menyaring berbagai komentar berpotensi cyberbullying dalam konten blog. Dalam penelitian ini akan membahas bagaimana mengembangkan sistem filter komentar cyberbullying pada konten blog yang nantinya dapat digunakan untuk meminimalisir penyalahgunaan komentar, khususnya terkait bullying oleh pengguna melalui fitur komentar pada konten blog. Cyberbullying merupakan ancaman nyata dalam interaksi di antara penulis konten blog dan pembaca artikel blog. Cyberbullying berdampak negatif terhadap korbannya seperti gangguan mental, tekanan emosional, hingga trauma sosial. Penelitian ini membahas bagaimana pengembangan fitur penyaring komentar cyberbullying pada konten blog untuk meminimalisir cyberbullying dalam suatu situs blog.

II. TINJAUAN PUSTAKA

Pada bagian ini dijelaskan tentang teori penunjang yang digunakan dalam pengembangan sistem ini meliputi:

A. CYBERBULLYING

Menurut Williard (2005), *cyberbullying* adalah perlakuan kejam yang dilakukan dengan sengaja kepada orang lain dengan mengirimkan atau mengedarkan materi berbahaya atau terlibat dalam bentuk agresi sosial menggunakan internet atau teknologi digital lainnya. Adapun aspek-aspek *cyberbullying* meliputi [2]:

1. *Flamming* yakni perilaku pengiriman pesan teks dengan kata-kata kasar dan frontal.
2. *Harassment* yakni perilaku pengiriman pesan tidak sopan kepada seseorang berupa gangguan yang dikirimkan melalui email, sms, atau pesan singkat di jaringan media sosial secara terus menerus.
3. *Denigration* yakni perilaku mengubar keburukan seorang di internet dengan maksud merusak reputasi dan nama baik dari orang yang dituju..
4. *Impersonation* yakni perilaku berpura-pura menjadi orang lain dan mengirimkan pesan yang tidak baik.
5. *Outing and Trickery* yakni perilaku menyebarkan rahasia orang lain atau foto pribadi orang lain.
6. *Exclusion* yakni perilaku yang dengan sengaja dan kejam menghilangkan orang dari grup online
7. *Cyberstalking* yakni perilaku berulang kali mengirimkan ancaman berbahaya atau pesan yang mengintimidasi menggunakan komunikasi elektronik.

Dalam penelitian ini lebih difokuskan pada aspek-aspek *cyberbullying* meliputi *flamming, harassment, denigration, impersonation, dan cyberstalking*.

B. NAÏVE BAYES CLASSIFIER

Naive Bayes Classifier (NBC) merupakan metode pembelajaran dengan konsep probabilitas sederhana [3]-[5]. NBC menggunakan teorema kuno, warisan abad ke-18, yang ditemukan oleh Thomas Bayes. NBC menyertakan dokumen klasifikasi terbimbing, metode pembelajaran yang menghasilkan fungsi untuk memetakan masukan ke keluaran yang diinginkan. NBC menganggap kemunculan satu kata tidak mempengaruhi kemunculan kata lainnya. NBC mampu memberikan kinerja yang cukup baik untuk banyak kasus modern dengan data yang besar. Adapun untuk menghitung probabilitas fitur kata menggunakan persamaan (1):

$$\hat{P}(w \mid c) = \frac{count(w,c)+1}{count(c)+|V|} \quad (1)$$

Kemudian untuk menghitung probabilitas prior menggunakan persamaan (2):

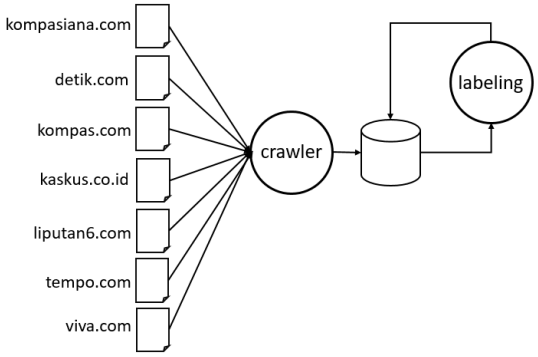
$$\hat{P}(c) = \frac{N_c}{N} \quad (2)$$

Terakhir untuk menentukan sentimen menggunakan persamaan (3):

$$c = \text{argmax}_c \hat{P}(c) \prod_i \hat{P}(w_i \mid c) \quad (3)$$

C. DATASET

Dataset yang digunakan dalam penelitian ini adalah pasangan data berita dan komentar-komentarnya yang diperoleh melalui proses crawling sejumlah website berita di Indonesia, yaitu kompasiana, detik, dan kompas, viva, tempo, liputan6, serta sebuah website forum, yaitu kaskus, seperti yang ditunjukkan pada gambar 1 berikut ini.



GAMBAR 1. Pembentukan Dataset

Kepada semua laman website yang menjadi sumber data, dilakukan pembatasan topik yang diambil. Adapun topik yang diolah hanyalah topik sosial budaya, politik, dan olahraga. Ketiga topik tersebut dipilih karena biasanya merupakan diskusi sensitif dan rawan *cyberbullying* di kalangan pembacanya. Total artikel yang diperoleh untuk dataset ini adalah 686 artikel dari berbagai sumber. Sedangkan untuk komentar terdiri dari 9803 komentar dari 686 artikel. Setiap komentar tersebut selanjutnya akan diberi label secara manual. Proses pelabelan dilakukan untuk membedakan komentar menjadi dua sentimen yaitu *cyberbullying* dan non-*cyberbullying*. Pada akhir proses pembentukan dataset, diperoleh 4.598 komentar yang termasuk dalam kategori *cyberbullying* dan 5205 komentar yang termasuk dalam kategori non-*cyberbullying*. Pasangan data artikel dan komentar-komentarnya yang diperoleh dari proses crawling tersebut selanjutnya akan disimpan ke dalam database [6] untuk kemudian dilanjutkan dengan preprocessing. Tahap preprocessing itu sendiri akan dijelaskan pada subbab yang terpisah.

D. PREPROCESSING

Preprocessing merupakan memproses data uji sebelum digunakan dalam program bertujuan untuk mengurangi jumlah kosa kata, menyeragamkan kata dan menghilangkan *noise* [7]. Setiap tahapan yang dilakukan dalam *preprocessing* adalah sebagai berikut:

1. *Case folding* adalah proses mengubah huruf dalam dokumen menjadi huruf kecil. Dokumen mengandung beragam variasi bentuk huruf sampai tanda baca. Variasi huruf ini harus diseragamkan dan tanda bacanya harus dihilangkan. Hanya huruf 'a' sampai dengan 'z' yang diterima. Karakter lainnya dihilangkan dan dianggap sebagai delimiter.
2. *Cleaning* yakni proses menghilangkan dokumen dari mention, hastag, link, emoticon, dan karakter lainnya yang tak berguna.
3. *Stopword* yakni proses menghapus kata-kata yang tidak perlu dari dokumen. Pada tahap *stopword-list*, kata-kata yang tidak penting dibuang dari daftar kata, misalnya kata "yang", "dimana", "mengapa", "yaitu", "yakni", dan sebagainya.
4. Normalisasi bahasa gaul adalah proses mengubah kata-kata tak lazim berupa kata-kata gaul menjadi kata-kata formal berbahasa Indonesia. Umumnya, tidak semua komentar pada suatu artikel menggunakan bahasa formal. Pembaca biasanya juga memakai bahasa gaul, seperti nggak, gue, loe, dll. Kata-kata yang tak lazim ini perlu diseragamkan melalui normalisasi bahasa menjadi bahasa formal berbahasa Indonesia.
5. *Stemming* adalah proses mengubah kata menjadi kata dasar. Pada umumnya kata dalam dokumen memiliki variasi kombinasi imbuhan kata yang beragam, seperti imbuhan awalan, akhiran, sisipan, dan kombinasi. Kata-kata tersebut perlu diseragamkan menjadi kata dasar supaya seragam dan mengurangi komplesitas kata. Adapun algoritma stemming yang terkenal ialah algoritma Nazief dan Adriani. Algoritma ini dikembangkan berdasarkan aturan morfologi Bahasa Indonesia yang mengelompokkan imbuhan menjadi awalan (prefiks), sisipan (infiks), dan akhiran (suffiks) dan gabungan awalan-akhiran (confiks). Algoritma ini menggunakan kamus kata dasar dan mendukung *recording*, yakni penyusunan kembali kata-kata yang mengalami proses *stemming* berlebih.
6. *Tokenizing* adalah proses pemotongan dokumen menjadi kata-kata setelah menjadi proses *filtering*. Hasil pemotongan kata-kata tersebut dijadikan kumpulan kata dan membentuk daftar kata. Potongan tersebut dikenal dengan istilah token

III. METODE

Perancangan sistem ini menggunakan metodologi *iterative waterfall* yang terdiri atas empat tahapan meliputi:

1. Analisis sistem yakni tahapan analisis terhadap kebutuhan sistem. Untuk itu, diperlukan sejumlah literatur terkait berupa jurnal dan buku-buku yang relavan terkait pengembangan sistem manajemen konten dan analisis sentimen *cyberbullying* guna mendapat informasi terkait kebutuhan, fitur dan batasan dalam pengembangan sistem.
2. Desain sistem yakni tahapan desain sistem berupa perancangan arsitektur, database, interface, dan desain prosedural sesuai dengan kebutuhan dan fitur-fitur yang akan dikembangkan.
3. Implementasi yakni tahapan implementasi sistem dalam terhadap modul-modul yang dikembangkan dengan cara pemograman. Pengembangan CMS dalam penelitian ini menggunakan bahasa PHP menggunakan *framework* CodeIgniter.
4. Pengujian yakni tahapan pengujian terhadap sistem baik berupa pengujian fungsional maupun non fungsional, guna mengetahui bahwa sistem yang dikembangkan dapat berjalan secara baik. Adapun pengujian fungsional sistem menggunakan *black box testing.* Sedangkan untuk mengetahui tingkat keakuratan sistem dalam mengklasifikasikan sentimen *cyberbullying* digunakan uji *hold-out* dimana dataset dibagi menjadi dataset latih dan dataset testing. Adapun model akan dievaluasi menggunakan parameter *accuracy and error rate*

IV. HASIL EKSPERIMEN DAN PENELITIAN

Pada bagian ini akan dijelaskan tentang hasil eksperimen dan penelitian. Adapun penjelasannya meliputi:

A. FITUR-FITUR SISTEM

Sistem CMS dalam penelitian ini memiliki beberapa fitur yang terbagi atas fitur front end dan fitur back end. Sistem ini memiliki 3 level hak akses meliputi pembaca, member, dan admin dimana masing-masing user memiliki hak akses berbeda. Selengkapnya ditunjukan pada tabel I

TABEL I
FITUR FRONT END

	Fitur	Pembaca	Member	Admin
Front End				
1.	Registrasi	V	X	X
2.	Login	X	V	V
3.	Artikel	V	V	V
4.	Komentar artikel	X	V	V
5.	Rating artikel	X	V	V
6.	Statistik artikel	X	V	V
7.	Kategori artikel	V	V	V
8.	Following	X	V	V
9.	Peringatan *cyberbullying*	X	V	V
10.	Pelaporan Komentar *cyberbullying*	X	V	V
11.	Tag Terpopuler	V	V	V
12.	Artikel Terpopuler	V	V	V
13.	Berbagi ke sosial media	V	V	V
14.	Tentang web	V	V	V
15.	Kebijakan web	V	V	V
16.	Evaluasi web	X	V	V
17.	Profil	V	V	V

. Selanjutnya CMS juga memiliki fitur *backend*. Pada bagian *backend* terdapat 2 level hak akses yakni member dan admin. Pada bagian *backend*, pembaca tidak bisa

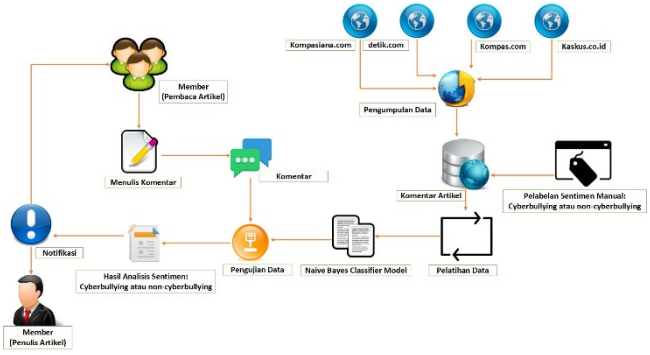
mengaksesnya. Selengkapnya ditunjukan pada tabel II di bawah ini:.

TABEL II
FITUR BACK END

	Fitur	Pembaca	Member	Admin
Back End				
1.	Konfigurasi profil	X	V	V
2.	Manajemen user	X	X	V
3.	Notifikasi	X	V	V
4.	Mengelola artikel	X	V	V
5.	Mengelola follower-following	X	V	V
6.	Mengelola rating	X	V	V
7.	Mengelola komentar	X	V	V
8.	Manajemen penyaring komentar *cyberbullying*	X	V	V
9.	Mengelola kategori	X	X	V
10.	Manajemen galeri	X	V	V
11.	Manajemen tag	X	X	V
12.	Manajemen tentang web	X	X	V
13.	Manajemen kebijakan	X	X	V
14.	Manajamen evaluasi web	X	X	V

B. ARSITEKTUR APLIKASI PENYARING KOMENTAR CYBERBULLYING

Pada gambar 2, ditunjukkan arsitektur penyaring komentar *cyberbullying* beserta cara kerjanya. Pertama, dataset komentar diambil secara manual dari berbagai sumber data seperti detik.com, kompas.com, dll. Dataset komentar selanjutnya disimpan ke database. Setelah setiap dataset yang ada dilabeli secara manual, dataset komentar dibagi menjadi dataset training dan dataset test.

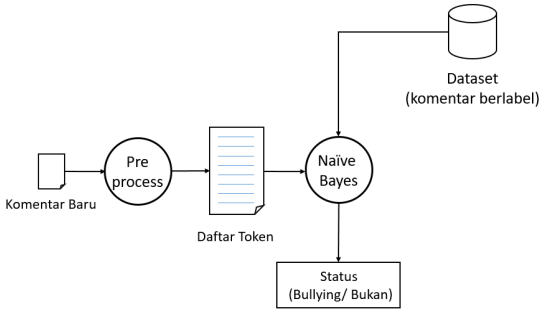


GAMBAR 2. **Arsitektur Aplikasi Penyaring Komentar Cyberbullying**

Dataset pelatihan dilatih untuk menghasilkan *Naive Bayes Classifier Model* [8]. Model tersebut digunakan untuk menyaring komentar dari pembaca yang mengirimkan komentar terkait konten blog. Kemudian, model akan mendeteksi komentar yang mengandung *cyberbullying*, kemudian sistem mengirimkan notifikasi kepada pembuat artikel dan pembaca artikel tentang peringatan *cyberbullying*[9][10].

C. ARSITEKTUR PENDETEKSI CYBERBULLYING

Seperti yang ditunjukkan pada gambar 3 mengenai arsitektur pendeteksi *cyberbullying,* pada bagian ini akan dijelaskan arsitektur beserta cara kerjanya pendeteksian *cyberbullying*. Pertama, kepada setiap komentar baru akan dilakukan preprocessing yang sama seperti pada tahap pembentukan dataset. Setelah diperoleh setiap token yang merupakan fitur dari komentar tersebut, akan dilakukan perhitungan probabilitas prior dan posterior dengan menggunakan Naïve Bayes Classifier. Dengan menggunakan model Naïve Bayes Classifier yang telah ditraining sebelumnya dengan menggunakan dataset yang ada, maka komentar yang baru akan diuji sentimennya oleh sistem dengan menghitung Vmap dan menentukan sentimen komentar tersebut [6][7]. Dalam penghitungan Vmap, semua kata komentar akan diberi bobot untuk menghasilkan sentimen *cyberbullying* dan non-*cyberbullying* sentimen. Jika bobot sentimen *cyberbullying* lebih besar dari bobot sentimen non *cyberbullying*, maka dapat disimpulkan bahwa komentar tersebut termasuk dalam *cyberbullying*. Jika bobot sentimen *cyberbullying* tidak lebih besar dari bobot sentimen non *cyberbullying* maka dapat disimpulkan bahwa komentar tersebut adalah non *cyberbullying*. Terakhir, hasilnya adalah sentimen *cyberbullying* atau sentimen non-*cyberbullying* dari komentar anggota.



GAMBAR 3. **Arsitektur Penyaring Komentar Cyberbullying**

D. PENGUJIAN

Pengujian terhadap penyaring komentar *cyberbullying* dilakukan dengan dua mode yakni pengujian menggunakan data pelatihan dan pengujian menggunakan data testing. Berdasarkan pengujian model dengan mode data pelatihan menggunakan total 7755 dataset komentar, maka dapat disimpulkan bahwa model memiliki akurasi sebesar 85,25% dan kesalahan 14,75%. Selengkapnya hasil pengujian menggunakan data pelatihan dapat dilihat pada gambar 4.

GAMBAR 4. Pengujian Model Menggunakan Data Pelatihan

Sedangkan berdasarkan pengujian model dengan mode data testing menggunakan total 1936 komentar dataset, didapatkan hasil model memiliki akurasi sebesar 80,48% dan kesalahan 19,52%

V. KESIMPULAN

Berdasarkan pembahasan di atas, dapat disimpulkan:

1. Dengan adanya aplikasi penyaringan komentar pada aplikasi web, dapat membantu menghindarkan masyarakat dari potensi *cyberbullying*.
2. Pengembangan fitur penyaring komentar *cyberbullying* menggunakan naive bayes classifier menghasilkan rata-rata akurasi sebesar 80% dan rata-rata error sebesar 20%.
3. Penggunaan algoritma Naïve Bayes untuk proses klasifikasi tergolong sederhana dan menuntut jumlah dataset yang besar, serta pembagian jumlah dataset untuk setiap class yang seimbang untuk dapat menghasilkan tingkat akurasi yang cukup baik.

PERAN PENULIS

Setiap penulis memiliki kontribusi yang sama dalam Analisis Formal, Investigasi, Administrasi Proyek, Sumber Daya, Perangkat Lunak, Validasi, Visualisasi, Penulisan dan Penyusunan Draf Asli.

COPYRIGHT

DAFTAR PUSTAKA

[1] Subagia, Anton. 2018. *Kolaborasi Codeigneter dan Ajax dalam Perancangan CMS*. Jakarta: PT Elex Media Komputindo.

[2] S. Hinduja & J. Patchin. 2010. *Bullying, Cyberbullying, adn Suicide*. Archives of Suicides Research. Vol. 14.

[3] Sipayung, M. Evasaria, dkk. 2016. *Perancangan Sistem Analisis Sentimen Komentar Pelanggan Menggunakan Metode Naive Bayes Classfier*. Jurnal Sistem Informasi (JSI). Vol. 8. No. 1. April 2016.

[4] Rahayu, Dwi Yeni Made Ni. 2018. *Rancangan Penerapan Metode Naive Bayes dalam Mendeteksi Hate Speech di Media Sosial*. Prosiding Seminar Nasional Pendidikan Teknik Informatika (SENTAPATI). Vol. 9. 8 September 2018.

[5] Kim Schouten , Onne van der Weijde, FlaviusFrasincar ,Rommert Dekker. 2018. *Supervised and Unsupervised Aspect Category detection for sentiment analysis with co-occurrence data.* IEEE Transactions on Cybernetics. Vol. 48, No. 4.

[6] Indrajani. 2014. *Pengantar Sistem Basis Data Case Study All In One*. Jakarta : PT. Elex Media Komputindo.

[7] Suyanto. 2018. *Machine Learning: Tingkat Dasar dan Lanjut.* Bandung: Informatika Bandung.

[8] Prabha, Surya, B Subbulakshmi. 2019. *Sentimental Analysis using Naive Bayes Classifier.* International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN).

[9] VandanaJha, Savitha.R ,P.DeepaShenoy, ArunKumarSangaiah, Venugopal KR. 2018. *A novel sentiment aware dictionary for multi domain sentiment classification*. Journal of Computers and Electrical Engineering. Vol. 69. Halm. 585-597.

[10] ShufengXiong , KuiyiWang, Donghong , Ji BingkunWang. 2018. *A short text sentiment topic model for product reviews*. Journal of Neurocomputing. Vol. 297. Halm. 94-102.

INSYST

Journal of Intelligent System and Computation

Volume 04 Nomor 02 Oktober 2022

Author Guidelines

- Manuscript should be written in Indonesia and be submitted online via journal website. Online Submission will be charged at no Cost
- Manuscript should not exceed 15 pages including embedded figures and tables, without any appendix, and the file should be in Microsoft Office (.doc/.docx). download template
- Title should be less than 15 words
- Abstracts consists of no more than 200 words, contains the essence of the article and includes a brief background, objectives, methods and results or findings of the study. Abstract is written in one paragraph.
- Keywords are written in Indonesia three to five words/phrases, separated with coma and consist of important words/phrases from the article.
- Author's name, affiliation, affiliation address and email. State clearly and include country's name on your affiliation address.
- The main text of the writing should be consists of: Introduction, Method, Result and Discussion, and Conclusion; followed by Acknowledgment and Reference
- Introduction State adequate background, issues and objectives, avoiding a detailed literature survey or a summary of the results. Explain how you addressed the problem and clearly state the aims of your study.
- Used method is the scientific in the form of study of literature, observation, surveys, interviews, Focus Group Discussion, system testing or simulation and other techniques commonly used in the world of research. It is also recommended to describe analysis techniques used briefly and clearly, so that the reader can easily understand.
- Results should be clear, concise and not in the form of raw data. Discussion should explore the significance of the results of the work, not repeat them. Avoid extensive citations and discussion of published literature. INSYST will do the final formatting of your paper.
- Conclusion should lead the reader to important matter of the paper. Authors are allowed to include suggestion or recommendation in this section. Write conclusion, suggestion and/or recommendation in narrative form (avoid of using bulleting and numbering)
- Acknowledgments. It is highly recommended to acknowledge a person and/or organizations helping author(s) in many ways. Sponsor and financial support acknowledgments should be included in this section. Should you have lots of parties

to be acknowledged, state your acknowledgments only in one paragraph. Avoid of using bulleting and numbering in this section

- The number of references are not less than 10 with at least 8 primary references. Primary references are include journal, thesis, disertation and all kinds of research reports. All refferences must come from source published in last 7 years.
- Figure and table should be in black and white, and if it is made in color, it should be readable when it is later printed in black and white.
- Figure and table should be clearly readable and in a proportional measure to the overall page.

Tim Redaksi

Journal of Intelligent System and Computation

Departement of Informatics

Institut Sains dan Teknologi Terpadu Surabaya

Jl. Ngagel Jaya Tengah 73-77 Surabaya

Email: insyst@istts.ac.id

Website: https://jurnal.stts.edu/index.php/INSYST/index