

## Peningkatan Kinerja Analisis Sentimen pada Ulasan Aplikasi Identitas Kependudukan Digital (IKD) di Indonesia Menggunakan Algoritma Support Vector Machine (SVM) dan SMOTE

M. Rafly Gusmansyah<sup>1</sup>, Heri Hendrawan<sup>2</sup>, Rahmaddeni<sup>3</sup>, Rohid<sup>4</sup>

<sup>1,2,3,4</sup>Prodi Teknik Informatika, Universitas Sains dan Teknologi Indonesia Pekanbaru, Indonesia

E-Mail: <sup>1</sup>raflygusmansyah5@gmail.com, <sup>2</sup>herihendrawan86@gmail.com, <sup>3</sup>rahmaddeni@usti.ac.id, <sup>4</sup>rohidpku01@gmail.com

---

### Abstrak

Analisis sentimen terhadap ulasan aplikasi Identitas Kependudukan Digital di Indonesia memberikan wawasan penting bagi pengembang dalam meningkatkan kualitas aplikasi. Penelitian ini menggunakan algoritma Support Vector Machine (SVM), yang dikenal andal dalam menangani data berdimensi tinggi, untuk mengklasifikasikan sentimen ulasan pengguna. Untuk mengatasi ketidakseimbangan data, diterapkan metode Synthetic Minority Oversampling Technique (SMOTE). Data ulasan diperoleh melalui data crawling dari Google Play Store sebanyak 1.498 komentar, kemudian diproses melalui pembersihan teks, normalisasi, tokenisasi, stemming, dan representasi numerik menggunakan Word2Vec. Hasil penelitian menunjukkan bahwa mayoritas komentar bersifat netral (78,2%), sedangkan komentar positif dan negatif masing-masing sebesar 10,3% dan 6,3%. Model SVM tanpa SMOTE mencapai akurasi 89,00% dengan F1-Score macro average 0,87 dan weighted average 0,89. Setelah penerapan SMOTE, akurasi meningkat menjadi 91,00%, dengan F1-Score macro average dan weighted average sebesar 0,91. Perbaikan signifikan terjadi pada kelas netral, di mana F1-Score meningkat dari 0,81 menjadi 0,99, meskipun performa kelas positif dan negatif sedikit menurun. Hasil ini menunjukkan bahwa kombinasi SVM dan SMOTE efektif dalam meningkatkan akurasi dan efisiensi klasifikasi sentimen pada data tidak seimbang.

**Kata kunci:** analisis sentimen, identitas kependudukan digital, SMOTE, support vector machine (SVM), word2vec.

---

### Abstract

Sentiment analysis of user reviews on Indonesia's Digital Identity application provides valuable insights for developers to improve application quality. This study employs the Support Vector Machine (SVM) algorithm, known for its reliability in handling high-dimensional data, to classify user sentiment. To address data imbalance, the Synthetic Minority Oversampling Technique (SMOTE) is applied. A total of 1,498 user comments were collected through data crawling from the Google Play Store and processed through text cleaning, normalization, tokenization, stemming, and numerical representation using Word2Vec. The results show that the majority of comments are neutral (78.2%), while positive and negative comments account for 10.3% and 6.3%, respectively. The SVM model without SMOTE achieved an accuracy of 89.00%, with a macro average F1-Score of 0.87 and a weighted average of 0.89. After applying SMOTE, accuracy improved to 91.00%, with both macro and weighted average F1-Scores reaching 0.91. A significant improvement occurred in the neutral class, where the F1-Score increased from 0.81 to 0.99, although the performance of the positive and negative classes slightly declined. These findings indicate that the combination of SVM and SMOTE is effective in enhancing the accuracy and efficiency of sentiment classification on imbalanced data.

**Keywords:** digital identity, sentiment analysis, SMOTE, support vector machine (SVM), word2vec.

---

*This work is an open access article and licensed under a Creative Commons Attribution-NonCommercial ShareAlike 4.0 International (CC BY-NC-SA 4.0)*



## 1. PENDAHULUAN

Perkembangan teknologi digital di Indonesia telah mendorong transformasi berbagai layanan publik, salah satunya adalah pengelolaan identitas kependudukan melalui aplikasi digital. Salah satu inovasi dari pemerintah Indonesia dalam hal ini adalah aplikasi Identitas Kependudukan Digital (IKD) yang diluncurkan oleh Ditjen Dukcapil. Hingga akhir tahun 2023, tercatat lebih dari 6 juta pengguna telah mengaktifkan IKD di berbagai wilayah Indonesia, menurut data Kementerian Dalam Negeri.

Namun, meskipun jumlah pengguna meningkat, keluhan terhadap performa aplikasi masih sering ditemukan, seperti antarmuka yang tidak intuitif, gangguan teknis, dan keterbatasan fitur. Hal ini menunjukkan adanya gap antara adopsi dan kepuasan pengguna yang perlu ditangani melalui pendekatan berbasis data.

Ulasan pengguna di platform seperti Google Play Store menjadi sumber informasi yang sangat berharga untuk mengidentifikasi dan mengevaluasi persepsi serta pengalaman pengguna terhadap aplikasi ini. Namun demikian, ulasan tersebut bersifat tidak terstruktur dan umumnya mengalami ketidakseimbangan distribusi kelas sentimen, di mana ulasan netral atau positif cenderung lebih banyak daripada ulasan negatif yang menyebabkan penurunan akurasi dalam proses klasifikasi sentimen menggunakan metode konvensional.

Untuk mengatasi masalah tersebut, penelitian ini mengusulkan penerapan algoritma *machine learning* menjadi solusi yang menjanjikan. Algoritma *Support Vector Machine* (SVM), yang dikenal karena keandalannya dalam menangani data teks berdimensi tinggi, digunakan untuk klasifikasi sentimen[1]. Selain itu, untuk mengatasi ketidakseimbangan data sentimen, diterapkan pula teknik SMOTE (*Synthetic Minority Oversampling Technique*) untuk meningkatkan jumlah sampel pada kelas minoritas secara sintesis[2]. Pendekatan ini bertujuan untuk meningkatkan performa model dalam mengidentifikasi sentimen negatif dan netral tanpa mengurangi akurasi sentimen positif.

Berbagai penelitian sebelumnya telah mengeksplorasi pendekatan berbeda untuk analisis sentimen. Misalnya [2][3] yang menunjukkan bahwa kombinasi SVM dan SMOTE mampu meningkatkan akurasi klasifikasi pada data tidak seimbang. Namun, penelitian-penelitian tersebut sebagian besar berfokus pada domain berita atau opini publik, bukan pada ulasan aplikasi layanan publik resmi. Pada [4][5] dikatakan bahwa penggunaan SMOTE dapat meningkatkan nilai model dari presisi namun menurunkan nilai dari Recall, Lalu [6] menerapkan *k-fold cross validation Support Vector Machine-SMOTE* karena memiliki nilai AUC paling tinggi. Pada penelitian membandingkan beberapa metode untuk klasifikasi pada [7][8][9] *Support Vector Machine* memiliki performa yang baik. Meskipun penelitian-penelitian ini telah menunjukkan keberhasilan pada domain mereka masing-masing, penerapan SVM dengan SMOTE pada analisis sentimen ulasan aplikasi identitas kependudukan digital masih jarang dijelajahi.

Namun, pada penelitian sebelumnya [10], telah dilakukan analisis sentimen aplikasi Identitas Kependudukan Digital (IKD) menggunakan metode *Support Vector Machine* (SVM) tanpa adanya penanganan ketidakseimbangan data. Kebaruan dari penelitian ini terletak pada penerapan kombinasi SVM dan SMOTE dalam domain aplikasi identitas kependudukan digital, yang datanya memiliki karakteristik tersendiri berupa teks pendek, bahasa informal, dan konteks pelayanan publik. Selain itu, digunakan representasi vektor berbasis *Word2Vec* yang mempertimbangkan konteks semantik kata dalam teks, berbeda dari pendekatan tradisional berbasis *bag-of-words* atau TF-IDF. Dengan pendekatan ini, penelitian ini tidak hanya memperbaiki performa model dalam menangani data tidak seimbang, tetapi juga memberikan kontribusi metodologis dalam analisis sentimen berbasis *machine learning* pada layanan publik digital di Indonesia. Pada Penelitian [11][12][13] Pengkombinasian SVM dengan *Word2Vec* dinilai memiliki kualitas kinerja yang baik.

Pendekatan kombinasi antara algoritma *Support Vector Machine* (SVM) dan metode *Synthetic Minority Oversampling Technique* (SMOTE) ini bertujuan untuk meningkatkan performa model dalam menangani ketidakseimbangan data, tetapi juga menawarkan kontribusi baru dalam metodologi analisis sentimen dengan memanfaatkan representasi teks yang lebih kontekstual. Hasil penelitian diharapkan dapat memberikan wawasan baru yang berharga bagi pengembang aplikasi layanan publik digital di Indonesia dalam meningkatkan kualitas layanan mereka melalui optimalisasi teknologi *machine learning*.

## 2. METODE PENELITIAN

Analisis sentimen adalah bagian dari ilmu yang membantu dalam memahami perasaan orang-orang dengan melihat data yang sudah ada. *Data mining* merupakan metode yang digunakan untuk memanfaatkan informasi historis guna mendukung pengambilan keputusan yang lebih baik di masa depan. Dalam penelitian ini, Digunakan program yang bernama *Rapidminer* untuk membuat model dengan metode yang disebut SVM, Pada [14] menjelaskan bahwa *Rapidminer* adalah alat yang membantu kita mengelola data mining. Program ini menggunakan berbagai teknik untuk memprediksi dan menjelaskan informasi, sehingga pengguna bisa membuat keputusan yang tepat. Proses analisis

sentimen melibatkan pemahaman dan pengolahan data secara otomatis untuk menghasilkan informasi yang berguna. Penelitian [15] menjelaskan bahwa analisis sentimen adalah ilmu yang bisa menganalisis pendapat dan penilaian orang tentang suatu kejadian. Dalam penelitian ini, mereka melihat ulasan dari pengguna aplikasi IKD di Google Play Store.

Pendekatan pembelajaran mesin yang disebut Support Vector Machine (SVM) membagi data menjadi beberapa wilayah yang sesuai dengan setiap kelas menggunakan hyperplanes. Pendekatan ini banyak digunakan untuk tugas klasifikasi teks dan sering digunakan untuk kumpulan data besar, terutama yang bersumber dari internet. Membangun hyperplane dengan batas simetris dan menghindari terlalu dekat dengan salah satu kelas adalah ide dasar SVM. Untuk menjamin batas ideal antar kelas, prosedur ini dilakukan dengan mengukur margin dan mencari titik maksimal. Tujuan utama metode SVM adalah menentukan hyperplane yang optimal untuk digunakan sebagai pemisah kelas[16].

$$f(x) = w^T x + b = 0 \quad (1)$$

Dimana :

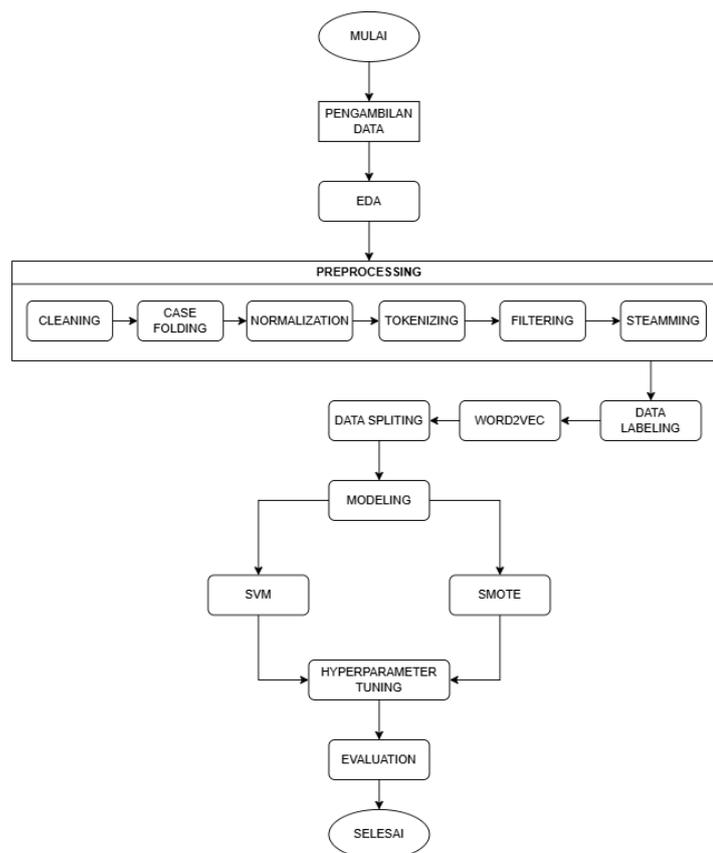
$w$  = vektor bobot (normal ke hyperplane)

$x$  = vektor data (fitur)

$b$  = bias (intersep dari hyperplane)

$f(x)$  = fungsi keputusan untuk menentukan kelas. [17]

Tahapan Penelitian yang dilakukan yaitu Pengumpulan Data (Data Crawling), EDA (Exploratory Data Analysis), Text Preprocessing, Modeling dan Model Evaluation. Tahapan penelitian yang dilakukan dalam analisis sentiment ini sebagai berikut :



Gambar 1. Alur Penelitian

Pada Gambar 1, alur penelitian diatas dapat dijelaskan sebagai berikut ;

### 2.1. Pengambilan Data

Data crawling merupakan metode yang digunakan untuk mengumpulkan data secara otomatis dari sumber-sumber eksternal seperti Google PlayStore[18]. Pengambilan data pada penelitian ini menggunakan teknik *web scrapping*, *Scraping* dilakukan menggunakan tools *Python* dan *library* seperti *BeautifulSoup* dan *Selenium* pada komentar pengguna aplikasi IKD di Playstore dan diambil sebanyak 1498 data komentar yang disimpan dalam format csv. Data ulasan yang digunakan dalam penelitian ini dikumpulkan dalam rentang waktu tahun 2023 hingga 2024, mencakup berbagai ulasan terbaru pengguna aplikasi Identitas Kependudukan Digital di Google Play Store. Pada tahap pengumpulan data dilakukan pemilihan atribut yang akan digunakan dalam penelitian ini. Terdapat empat atribut yaitu *username*, *score*, *at* dan *content*. Namun, atribut yang dipakai hanya atribut *content* karena berisi ulasan pengguna aplikasi IKD.

## 2.2. EDA

Selanjutnya, data yang telah diperoleh melalui scrapping akan dianalisis menggunakan Exploratory Data Analysis (EDA). Pada tahap ini, peneliti memeriksa distribusi data, mendeteksi nilai yang hilang, mendeteksi duplikasi data, serta mengevaluasi pola yang ada untuk memahami karakteristik data secara keseluruhan[19]. Proses ini penting sebelum melanjutkan ke tahap pemrosesan teks yang lebih dalam.

## 2.3. Preprocessing

Pada tahapan berikutnya, dilakukan text preprocessing, yang melibatkan serangkaian teknik untuk mempersiapkan data agar dapat digunakan dalam analisis lebih lanjut. Preprocessing data penting dalam analisis data mining untuk membersihkan, mengubah format, dan mempersiapkan data agar lebih mudah dan akurat[20]. Beberapa langkah yang diterapkan dalam text preprocessing antara lain adalah:

### 2.3.1. Data Cleaning

Pada titik ini, konten disaring untuk menghilangkan elemen asing termasuk URL, sebutan, hashtag, simbol khusus, angka, dan tanda baca yang berlebihan. Tujuan dari langkah ini adalah untuk jaminan bahwa hanya data relevan yang tersisa untuk prosedur berikut.

### 2.3.2. Case Folding

Mengubah semua huruf menjadi huruf kecil untuk menghindari perbedaan antara huruf besar dan kecil yang bisa mempengaruhi analisis.

### 2.3.3. Normalization

Untuk menyederhanakan data, seperti pengubahan angka atau simbol menjadi format yang konsisten.

### 2.3.4. Tokenizing

Pemecahan teks menjadi unit-unit terkecil yang disebut token.

Contoh : Input : "Aplikasi sangat membantu, Tapi sering error"  
Output : " ["aplikasi", "sangat", "membantu", "tapi", "sering", "error"] "

### 2.3.5. Stopword Removal

Menghapus kata-kata umum yang tidak memiliki kontribusi signifikan terhadap makna ("yang", "dan", "dengan").

Contoh : Input : ["aplikasi", "sangat", "membantu", "tapi", "sering", "error"]  
Output : " ["aplikasi", "membantu", "sering", "error"] "

### 2.3.6. Filtering

Untuk menghapus kata-kata yang tidak penting, seperti kata sambung atau kata yang terlalu sering muncul dalam data.

### 2.3.7. Stemming

Bertujuan untuk mengurangi kata-kata menjadi bentuk dasarnya, mengeliminasi variasi kata yang serupa.

Contoh : Input : ["aplikasi", "membantu", "sering", "error"]  
Output : ["aplikasi", "bantu", "sering", "error"]

## 2.4. Data Labeling

Setelah preprocessing, data diberi label dengan kategori Positif, Netral, dan Negatif. Tahapan ini penting untuk memastikan bahwa data memiliki anotasi yang diperlukan untuk proses supervised learning.

## 2.5. Word2vec

Data teks yang telah diproses kemudian diubah menjadi representasi numerik menggunakan Word2Vec. Teknik ini memungkinkan pemodelan semantik kata-kata berdasarkan konteksnya dalam teks.

### 2.5.1. Konfigurasi Word2Vec

Model: CBOW (Continuous Bag of Words) Cocok untuk dataset kecil hingga sedang dan menghasilkan vektor kata yang lebih stabil, Ukuran vektor (vector size): 100 Ukuran 100 umum digunakan dan cukup representatif, terutama untuk dataset ulasan dengan panjang kalimat sedang, Window size: 5, Ukuran 5 adalah nilai default dan sudah cukup baik untuk menangkap konteks lokal dalam kalimat, Minimum count: 2, Ini mencegah kata yang terlalu jarang muncul (noisy) masuk ke vektor (kata muncul minimal 2 kali), Epochs: 10 Untuk dataset dengan ukuran sedang, 10 epoch cukup untuk stabilitas tanpa overfitting, Word2Vec menghasilkan vektor representasi dari setiap kata dalam dataset, yang kemudian dirata-rata untuk setiap ulasan guna membentuk representasi fitur kalimat.

### 2.6. Splitting Data

Setelah data diproses, dilakukan pembagian data menjadi dua bagian: data pelatihan dan data pengujian. Data pelatihan digunakan untuk membangun model, sementara data pengujian digunakan untuk mengevaluasi kinerja model tersebut. Pembagian data ini penting untuk memastikan bahwa model tidak mengalami overfitting dan dapat memberikan hasil yang akurat ketika diterapkan pada data baru, Pembagian data dibagi menjadi 2 subset 80% untuk data *training* dan 20% untuk data *testing*.

### 2.7. SMOTE

Untuk menangani ketidakseimbangan kelas dalam data, tahap selanjutnya menggunakan teknik SMOTE (Synthetic Minority Over-sampling Technique), yang diperkenalkan oleh Chawla [21]. Teknik ini menghasilkan sampel sintetis dari kelas minoritas untuk mengimbangi jumlah data pada kelas mayoritas, dengan tujuan mengurangi bias pada model machine learning yang akan dibangun.

### 2.8. Modeling

Model machine learning dibangun menggunakan algoritma yang sesuai dengan permasalahan yang ingin diselesaikan. Proses ini melibatkan pemilihan fitur, parameter tuning, serta pelatihan model pada data pelatihan.

### 2.9. Hyperparameter Tuning

Penelitian ini menggunakan Support Vector Machine (SVM) untuk klasifikasi sentimen. Untuk optimasi model, dilakukan tuning hyperparameter dengan Grid Search Cross-Validation, menyesuaikan nilai C, kernel, dan gamma yang berpengaruh dalam pemisahan kelas sentimen.

### 2.10. Evaluasi Model

Model yang telah dilatih dievaluasi menggunakan data pengujian. Beberapa metrik evaluasi yang umum digunakan meliputi Akurasi, Presisi, Recall, F1-score dan Support. Tahapan ini bertujuan untuk memastikan bahwa model yang dihasilkan mampu memberikan kinerja yang optimal.

## 3. HASIL DAN PEMBAHASAN

Hasil dan pembahasan yang telah dilakukan akan disajikan lebih lanjut pada bagian hasil dan pembahasan ini, berdasarkan uraian yang telah diberikan pada tahapan penelitian di atas. Adapun tools yang digunakan dalam penelitian ini adalah Google Colab dan Python serta library pendukung lainnya.

### 3.1. Dataset

Pengambilan data pada penelitian ini menggunakan teknik web scrapping pada komentar pengguna aplikasi IKD di Google Playstore dan diambil sebanyak 1498 data komentar yang disimpan dalam format csv seperti yang terlampir pada Tabel 1 Dibawah ini yang memperlihatkan beberapa contoh ulasan mentah yang dikumpulkan dari Google Play Store. Komentar ini beragam dari segi panjang, struktur kalimat, serta penggunaan bahasa informal, yang menjadi tantangan dalam preprocessing dan klasifikasi..

Tabel 1 . Dataset hasil *web scrapping* dari Google Playstore

---

Daftar ke apk harus scan barcod dulu ke dukcapil. Nggak efisien dan efektif.

---

Ini gimana ini buka aplikasi ny, terjadi kesalahan koneksi trus pdhl sinyal sdh bagus.

---

Menurut saya, ini sangat bagus. Memudahkan masyarakat dalam menyimpan berkas identitas diri dan keluarga.

---

### 3.2. Exploratory Data Analysis (EDA)

Setelah data dibersihkan, selanjutnya dilakukan proses identifikasi pada data, tujuan utama dari proses ini adalah untuk melihat serta memastikan bahwa data teks yang akan diproses sudah sesuai untuk digunakan dalam analisis lanjutan. Dari proses ini didapatkanlah beberapa informasi mengenai data yang selanjutnya akan digunakan untuk analisis lebih lanjut kedalam model.

```

RangeIndex: 1498 entries, 0 to 1497
Data columns (total 20 columns):
 #   Column              Non-Null Count  Dtype
---  ---
 0   Komentar            1498 non-null   object
 1   Komentar.1         1498 non-null   object
 2   Komentar.2         1498 non-null   object
 3   Komentar.3         1498 non-null   object
 4   Komentar.4         1498 non-null   object
 5   Komentar.5         1498 non-null   object
 6   Komentar.6         1498 non-null   object
 7   Komentar.7         1498 non-null   object
 8   Komentar.8         1498 non-null   object
 9   Komentar.9         1498 non-null   object
10  Komentar.10        1498 non-null   object
11  Komentar.11        1498 non-null   object
12  Komentar.12        1498 non-null   object
13  Komentar.13        1498 non-null   object
14  Komentar.14        1498 non-null   object
15  Komentar.15        1498 non-null   object
16  Komentar.16        1498 non-null   object
17  Komentar.17        1498 non-null   object
18  text_length        1498 non-null   int64
19  tokens             1498 non-null   object
dtypes: int64(1), object(19)
memory usage: 234.2+ KB
    
```

Gambar 2. Hasil Dari EDA

Dari gambar 2 dapat diambil informasi :

- Dataset memiliki 1.498 baris dan 20 kolom.
- Terdapat 2 type data yang digunakan dalam setiap kolom, yaitu int64, dan Object.
- Tidak ada nilai yang hilang didalam dataset
- Ukuran dataset menggunakan memori yang relatif kecil yaitu 234.2 KB.

### 3.3. Pre-Processing

Cleaning, Case Folding, Normalization, Tokenization, Filtering dan Stemming adalah langkah lanjutan dalam proses ini. Data teks akan dibersihkan menjadi data yang siap untuk digunakan pada langkah selanjutnya. Tabel 2 di bawah memperlihatkan hasil dari proses preprocessing teks yang mencakup tokenisasi, stopwords removal, dan stemming. Proses ini menghasilkan teks yang bersih dan siap dikonversi menjadi vektor numerik menggunakan Word2Vec.

Tabel 2. Hasil dari proses *preprocessing*

Teks Awal	Hasil Preprocessing
Aplikasi Setengah-Setengah... Apa bedanya klo kita harus ke Capil dlu, sekalian aja offline dong...	"aplikasi", "setengah", "setengah", "beda", "kalau", "harus", "capil", "dulu", "sekalian", "offline"
Aplikasi sampah, tidak bisa di buka alasannya koneksi. Punya e ktp malahan aplikasinya lelet	"aplikasi", "sampah", "bisa", "buka", "alasan", "koneksi", "ktp", "malahan", "aplikasi", "lelet"

### 3.4. Data Labeling

Dalam penelitian ini, proses pelabelan data dilakukan menggunakan pendekatan lexicon-based sentiment analysis. Setiap ulasan yang diperoleh dari Google Play Store dianalisis berdasarkan daftar kata sentimen (lexicon) untuk menentukan kategori sentimen yang sesuai. Proses ini dilakukan dengan mencocokkan setiap kata dalam ulasan dengan kamus sentimen, yang berisi daftar kata dengan nilai sentimen tertentu. Jika ulasan mengandung lebih banyak kata dengan sentimen positif, maka akan diberi label positif. Sebaliknya, jika didominasi kata-kata negatif, maka ulasan diklasifikasikan sebagai negatif. Sementara itu, ulasan yang tidak memiliki kecenderungan kuat terhadap sentimen positif atau negatif dikategorikan sebagai netral.

### 3.5. Feature Weighting

Pembobotan fitur dilakukan menggunakan teknik Word2Vec untuk merepresentasikan kata dalam bentuk vektor numerik. Teknik ini digunakan untuk menangkap informasi penting dalam teks dengan mempertimbangkan hubungan semantik antar kata. Dengan pendekatan ini, setiap kata dalam dataset dikonversi menjadi vektor yang dapat digunakan sebagai fitur dalam proses klasifikasi seperti gambar 3 yang menunjukkan hasil visualisasi vektor kata yang dihasilkan oleh proses pelatihan Word2Vec.

	Kata	Vektor_1	Vektor_2	Vektor_3	Vektor_4	Vektor_5	Vektor_6
0	aplikasi	-0.125217	0.481397	-0.174161	0.137252	0.120267	-0.818477
1	ya	-0.154607	0.468151	-0.106442	0.074654	-0.017513	-0.856034
2	ktp	-0.357698	0.386145	-0.486944	0.074833	-0.289306	-0.690381
3	buka	0.002162	0.881816	-0.563407	-0.027080	0.738510	-1.040389
4	data	-0.083087	0.758304	-0.459695	-0.103630	0.229446	-0.951534

Gambar 3. Hasil Dari Proses Word2vec

### 3.6. Hyperparameter Tuning

Pada penelitian ini, proses klasifikasi sentimen menggunakan algoritma Support Vector Machine (SVM). Untuk memperoleh hasil klasifikasi yang optimal, dilakukan tuning hyperparameter menggunakan metode Grid Search Cross-Validation. Hyperparameter yang disesuaikan meliputi nilai C, jenis kernel, serta gamma, yang berperan dalam menentukan kinerja model dalam memisahkan kelas sentimen.

#### 3.6.1. Hyperparameter SVM Tanpa SMOTE

Dalam eksperimen awal tanpa teknik penyeimbangan data, SVM diuji dengan beberapa kombinasi parameter.

```
=== SVM TANPA SMOTE ===  
Best Params: {'C': 10, 'gamma': 'scale', 'kernel': 'linear'}  
Accuracy: 0.8400  
Precision: 0.8313  
Recall: 0.8400  
F1 Score: 0.8334
```

Gambar 4. Hasil optimasi model SVM

Pada Gambar 4, Hasil tuning menunjukkan bahwa kombinasi terbaik diperoleh dengan parameter:

- C = 1, yang mengontrol regulasi model dengan memastikan keseimbangan antara margin maksimal dan kesalahan klasifikasi.
- Kernel = "linear", yang menghasilkan pemisahan data dalam ruang fitur yang lebih baik untuk dataset ini.
- Gamma = "scale", yang memungkinkan model untuk menentukan pengaruh masing-masing titik data terhadap keputusan klasifikasi.

#### 3.6.2. Hyperparameter SVM dengan SMOTE

Untuk mengatasi ketidakseimbangan data dalam dataset ulasan, diterapkan Synthetic Minority Oversampling Technique (SMOTE) sebelum proses klasifikasi menggunakan SVM.

```
=== SVM DENGAN SMOTE ===  
Best Params: {'svm_kernel': 'linear', 'svm_gamma': 'auto', 'svm_C': 1}  
Accuracy: 0.8600  
Precision: 0.8512  
Recall: 0.8600  
F1 Score: 0.8414
```

Gambar 5. Hasil optimasi model SVM dengan SMOTE

Pada Gambar 5, Setelah dilakukan tuning hyperparameter, diperoleh kombinasi parameter terbaik sebagai berikut:

- $C = 1$ , yang memberikan keseimbangan optimal antara margin keputusan dan generalisasi model.
- Kernel = "linear", yang mempertahankan keakuratan model dalam klasifikasi data seimbang hasil SMOTE.
- Gamma = "auto", yang memungkinkan model untuk menyesuaikan pengaruh tiap data terhadap hyperplane secara otomatis.

Dengan konfigurasi ini, model SVM + SMOTE mencapai akurasi terbaik sebesar 86%, yang sedikit lebih tinggi dibandingkan model SVM tanpa SMOTE. Penerapan SMOTE membantu meningkatkan keseimbangan distribusi data sehingga model lebih efektif dalam mengklasifikasikan sentimen negatif dan positif.

### 3.7. Evaluasi Model

Evaluasi terhadap kedua model dilakukan menggunakan Confusion Matrix.

#### 3.7.1. Support Vector Machine (SVM)

Evaluasi model Support Vector Machine (SVM) dilakukan untuk mengukur performa algoritma dalam melakukan klasifikasi. Setelah proses pelatihan, model dievaluasi menggunakan confusion matrix, yang menampilkan metrik seperti precision, recall, dan F1-score untuk setiap kelas.

METRIKSI				
Accuracy Score untuk Support Vector Machine Model :: 0.8900487125956854				
	precision	recall	f1-score	support
negatif	0.95	0.90	0.93	721
netral	0.76	0.86	0.81	288
positif	0.89	0.89	0.89	428
accuracy			0.89	1437
macro avg	0.87	0.88	0.87	1437
weighted avg	0.89	0.89	0.89	1437

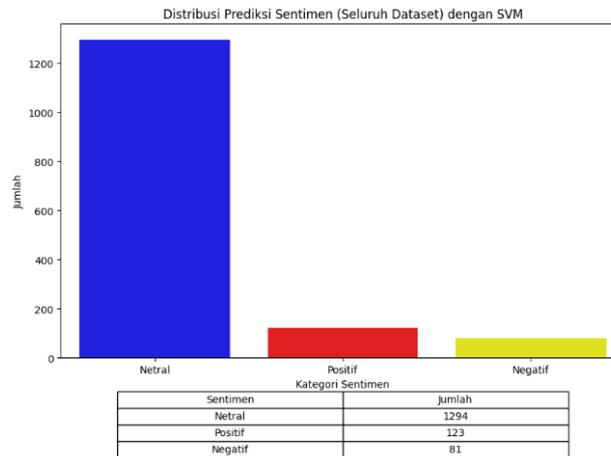
Gambar 6. Metric SVM Tanpa SMOTE

Dari Gambar 6 bisa dilihat Evaluasi model confusion metric Pada Algoritma Support Machine Vector memiliki nilai akurasi 89% dan menunjukkan performa model SVM sebelum penerapan SMOTE. Kelas netral memiliki skor F1 terendah (0.81), menunjukkan model kesulitan mengenali kelas tersebut secara akurat.

Tabel 3. Hasil skor evaluasi metrik

Metric	Positif	Negatif	Netral
<b>1</b> Precision	0.89	0.95	0.76
<b>2</b> Recall	0.89	0.90	0.85
<b>3</b> F1-Score	0.89	0.90	0.81
<b>4</b> Accuracy		0.89	

Pada Tabel 3 bisa dilihat Evaluasi model confusion metric Pada Algoritma Support Machine Vector memiliki nilai presisi pada positif 89% pada Negatif 95% dan Pada Netral 76%, nilai recall pada positif 89% pada Negatif 90% dan Pada Netral 85%, nilai f1-score pada positif 89% pada Negatif 90% dan Pada Netral 81%.



Gambar 7. Hasil Prediksi Klasifikasi Menggunakan SVM Tanpa SMOTE

Pada Gambar 7, Berdasarkan hasil analisis sentimen menggunakan algoritma Support Vector Machine (SVM) terhadap ulasan aplikasi Identitas Kependudukan Digital di Indonesia, distribusi prediksi sentimen menunjukkan bahwa sebagian besar komentar bersifat netral, dengan total 1.235 komentar. Sementara itu, terdapat 163 komentar positif dan 100 komentar negatif. Visualisasi dalam bentuk diagram batang menunjukkan dominasi sentimen netral dibandingkan dengan sentimen positif maupun negatif. Hal ini mengindikasikan bahwa sebagian besar pengguna memberikan ulasan tanpa ekspresi emosional yang kuat terhadap aplikasi tersebut.

### 3.7.2. SVM dengan SMOTE

Evaluasi model setelah penerapan SMOTE dilakukan untuk mengukur dampak teknik oversampling terhadap performa klasifikasi. Model dievaluasi menggunakan confusion matrix, yang menunjukkan metrik seperti precision, recall, dan F1-score untuk setiap kelas.

```

Accuracy Score untuk Support Vector Machine Model :: 0.9135338345864662
precision recall f1-score support
negatif    0.87    0.89    0.88    272
netral     0.99    1.00    0.99    266
positif    0.88    0.85    0.87    260

accuracy   0.91    0.91    0.91    798
macro avg  0.91    0.91    0.91    798
weighted avg 0.91    0.91    0.91    798
    
```

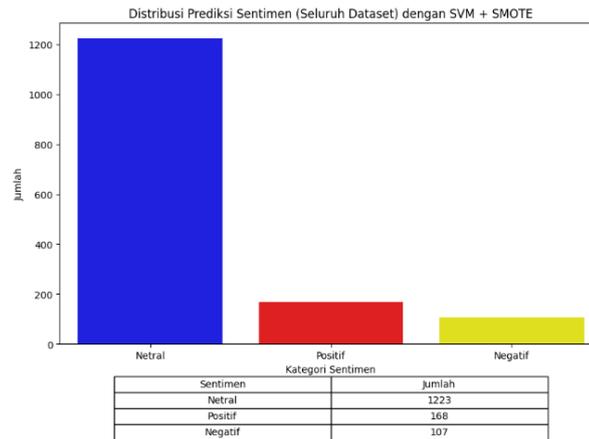
Gambar 8. Metric SVM dengan SMOTE

Dari Gambar 8 bisa dilihat Setelah SMOTE diterapkan, terjadi peningkatan akurasi dari 89% menjadi 91%. Perbaikan paling signifikan terdapat pada kelas netral dengan peningkatan F1-Score dari 0.81 menjadi 0.99. Ini menunjukkan bahwa SMOTE berhasil meningkatkan representasi kelas minoritas dalam proses pelatihan model..

Tabel 4. Hasil skor evaluasi metrik SMOTE

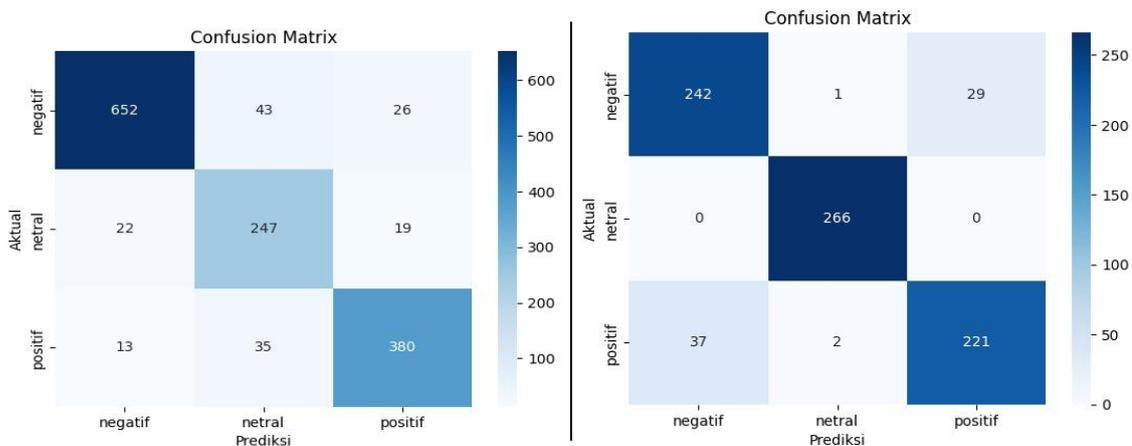
Metric	Positif	Negatif	Netral
<b>1</b> Precision	0.89	0.87	0.99
<b>2</b> Recall	0.85	0.89	1.00
<b>3</b> F1-Score	0.87	0.85	0.99
<b>4</b> Accuracy	0.91		

Pada Tabel 4 bisa dilihat Evaluasi model confusion matrix dengan penerapan SMOTE Pada Algoritma Support Machine Vector memiliki nilai presisi pada positif 89% pada Negatif 87% dan Pada Netral 99%, nilai recall pada positif 85% pada Negatif 89% dan Pada Netral 100%, nilai f1-score pada positif 87% pada Negatif 85% dan Pada Netral 99%.



Gambar 9. Hasil Prediksi Klasifikasi Menggunakan SVM Dengan SMOTE

Pada Gambar 9, Setelah menerapkan SMOTE untuk menangani ketidakseimbangan data, distribusi prediksi sentimen mengalami perubahan yang lebih seimbang. Jumlah komentar untuk kategori netral adalah 1.235, sementara kategori positif mencapai 163 dan kategori negatif berjumlah 100. Hal ini menunjukkan bahwa teknik SMOTE berhasil mengatasi bias distribusi yang sebelumnya didominasi oleh sentimen netral. Dari hasil visualisasi, terlihat bahwa jumlah sentimen positif (163) dan negatif (100) masih jauh lebih kecil dibandingkan netral (1.235). Meskipun begitu, penerapan SMOTE membantu meningkatkan representasi kategori minoritas, memungkinkan model untuk mengenali pola sentimen dari ketiga kategori secara lebih proporsional.



Gambar 10. Confusion Matrix SVM Tanpa SMOTE (kiri) dan Dengan SMOTE (kanan)

Dari hasil evaluasi model menggunakan Confusion Matrix pada gambar 10, SVM dengan penerapan SMOTE menunjukkan performa klasifikasi yang lebih konsisten dibandingkan dengan tanpa penerapan SMOTE, baik dalam hal presisi ataupun distribusi kesalahan.

Model SVM tanpa penerapan SMOTE menunjukkan akurasi sebesar 89,00%, dengan nilai F1-Score masing-masing sebesar 0,89 untuk kelas positif, 0,90 untuk kelas negatif, dan 0,81 untuk kelas netral. Penerapan SMOTE meningkatkan akurasi menjadi 91,00%, serta meningkatkan F1-Score kelas netral secara signifikan dari 0,81 menjadi 0,99. Namun, terjadi sedikit penurunan F1-Score pada kelas positif (dari 0,89 menjadi 0,87) dan kelas negatif (dari 0,90 menjadi 0,85). Hal ini menunjukkan bahwa SMOTE efektif dalam meningkatkan performa klasifikasi pada kelas yang sebelumnya under-represented.

Distribusi sentimen setelah diterapkannya model menunjukkan bahwa mayoritas ulasan bersifat netral (78,2%), sementara ulasan positif dan negatif masing-masing sebesar 10,3% dan 6,3%. Visualisasi hasil klasifikasi dan confusion matrix mendukung temuan ini, di mana distribusi prediksi menjadi lebih seimbang setelah penerapan SMOTE.

Pembobotan fitur menggunakan Word2Vec memberikan representasi semantik kata dalam bentuk vektor berdimensi 100. Representasi tersebut dirata-rata untuk setiap ulasan guna menghasilkan vektor

fitur tunggal sebagai input bagi model SVM. Visualisasi hasil Word2Vec menunjukkan bahwa kata-kata dengan makna serupa cenderung berada dalam kluster yang sama, yang mengindikasikan bahwa teknik ini mampu menangkap hubungan semantik antar kata secara efektif.

Meskipun hasil menunjukkan peningkatan performa model, penelitian ini memiliki beberapa keterbatasan. Data ulasan hanya diambil dari satu platform, yaitu Google Play Store, yang dapat membatasi keragaman opini pengguna. Proses pelabelan sentimen dilakukan secara otomatis menggunakan pendekatan leksikon, yang tidak selalu akurat dalam menangkap konteks semantik kompleks seperti sarkasme atau ironi. Selain itu, penggunaan rata-rata vektor Word2Vec untuk representasi ulasan tidak mempertimbangkan urutan kata, yang dapat berdampak pada kehilangan informasi struktural dalam kalimat.

Peluang pengembangan selanjutnya mencakup perluasan sumber data dari berbagai platform digital, penerapan metode pelabelan manual atau semi-otomatis untuk meningkatkan kualitas label, serta penggunaan model berbasis deep learning seperti BERT atau LSTM guna menangkap konteks kalimat yang lebih kompleks. Analisis lanjutan juga dapat mencakup dimensi temporal atau spasial untuk memahami dinamika perubahan sentimen dari waktu ke waktu atau antar wilayah.

#### 4. KESIMPULAN

Penelitian ini menunjukkan bahwa kombinasi algoritma Support Vector Machine (SVM) dan Synthetic Minority Oversampling Technique (SMOTE) efektif dalam meningkatkan performa klasifikasi sentimen pada ulasan aplikasi Identitas Kependudukan Digital (IKD) di Indonesia. Model SVM tanpa SMOTE mencapai akurasi 89%, dengan F1-Score tertinggi pada kelas negatif dan terendah pada kelas netral. Setelah penerapan SMOTE, akurasi meningkat menjadi 91%, disertai peningkatan signifikan F1-Score pada kelas netral, meskipun terjadi penurunan kecil pada kelas positif dan negatif.

Distribusi sentimen menunjukkan mayoritas ulasan bersifat netral (78,2%), sementara ulasan positif dan negatif masing-masing sebesar 10,3% dan 6,3%. Hasil ini menggambarkan bahwa sebagian besar pengguna menyampaikan ulasan tanpa ekspresi emosional yang kuat, meskipun tetap terdapat sejumlah ulasan negatif yang menunjukkan adanya aspek yang perlu diperhatikan.

#### DAFTAR PUSTAKA

- [1] Y. Wahba, N. Madhavji, and J. Steinbacher, "A Comparison of SVM against Pre-trained Language Models (PLMs) for Text Classification Tasks."
- [2] Abdul Hadi, B. Bernard, L. N. Hulu, and W. N. Hulu, "Sentimen Ulasan Pengguna E-Commerce Blibli Menggunakan Algoritma Naïve Bayes dan Support Vector Machine dengan Metode SMOTE," *Kohesi: Jurnal Multidisiplin Saintek*, vol. 6, no. 5, pp. 1–9, 2025, doi: 10.8734/Kohesi.v1i2.365.
- [3] Normah, B. Rifai, S. Vambudi, and R. Maulana, "Analisa Sentimen Perkembangan Vtuber Dengan Metode Support Vector Machine Berbasis SMOTE," *Jurnal Teknik Komputer AMIK BSI*, vol. 8, no. 2, pp. 174–180, 2022, doi: 10.31294/jtk.v4i2.
- [4] D. Fitria, T. H. Saragih, Muliadi, D. Kartini, and F. Indriani, "Classification of Appendicitis in Children Using SVM with KNN Imputation and SMOTE Approach to Improve Prediction Quality," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 6, no. 3, pp. 302–311, 2024, doi: 10.35882/jeeemi.v6i3.470.
- [5] W. P. Hutami, H. Wijayanto, and I. D. Sulvianti, "Penerapan Support Vector Machine dengan SMOTE Untuk Klasifikasi Sentimen Pemberitaan Omnibus Law Pada Situs CNNIndonesia.com," *Xplore: Journal of Statistics*, vol. 11, no. 1, pp. 26–35, 2022, doi: 10.29244/xplore.v11i1.852.
- [6] M. I. Putri and I. Kharisudin, "Penerapan Synthetic Minority Oversampling Technique (SMOTE) Terhadap Analisis Sentimen Data Review Pengguna Aplikasi Marketplace Tokopedia," *PRISMA, Prosiding Seminar Nasional Matematika*, vol. 5, pp. 759–766, 2022.
- [7] F. Putra, H. F. Tahiyat, R. M. Ihsan, R. Rahmaddeni, and L. Efrizoni, "Penerapan Algoritma K-Nearest Neighbor Menggunakan Wrapper Sebagai Preprocessing untuk Penentuan Keterangan Berat Badan Manusia," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 1, pp. 273–281, 2024, doi: 10.57152/malcom.v4i1.1085.

- [8] Zairi saputra, H. A. Supahri, R. Ismanizan, and R. Rahmaddeni, “Perbandingan Algoritma KNN (K-Nearest Neighbors), Naïve Bayes, Dan SVM (Support Vector Machine) Untuk Klasifikasi Pemberian Pinjaman Nasabah,” *Jurnal Ilmiah Sistem Informasi dan Teknik Informatika (JISTI)*, vol. 7, no. 1, pp. 67–75, 2024, doi: 10.57093/jisti.v7i1.182.
- [9] R. A. Husen, R. Astuti, L. Marlia, R. Rahmaddeni, and L. Efrizoni, “Analisis Sentimen Opini Publik pada Twitter Terhadap Bank BSI Menggunakan Algoritma Machine Learning,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 3, no. 2, pp. 211–218, 2023, doi: 10.57152/malcom.v3i2.901.
- [10] Nila Rusiardi Jayanti, “Analisis Sentimen Review Aplikasi Identitas Kependudukan Digital Menggunakan Algoritma Support Vector Machine,” *Global: Jurnal Lentera BITEP*, vol. 02, no. 04, pp. 3025–5503, 2024.
- [11] D. Utami, “Analisis Metode Wordtovec Pada Performa Klasifikasi Kategori Berita Menggunakan Metode SVM dan KNN,” *Explore*, vol. 11, no. 2, p. 82, 2021, doi: 10.35200/explore.v11i2.479.
- [12] M. M. Truşcă, “Efficiency of SVM classifier with Word2Vec and Doc2Vec models,” *Proceedings of the International Conference on Applied Statistics*, vol. 1, no. 1, pp. 496–503, 2019, doi: 10.2478/icas-2019-0043.
- [13] M. A. Fauzi, “Word2Vec model for sentiment analysis of product reviews in Indonesian language,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 1, p. 525, 2019, doi: 10.11591/ijece.v9i1.pp525-530.
- [14] W. Eko Saputro, H. Yuana, and W. Dwi Puspitarsari, “Analisis Sentimen Pengguna Dompot Digital Dana Pada Kolom Komentar Google Play Store Dengan Metode Klasifikasi Support Vector Machine,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 2, pp. 1151–1156, 2023, doi: 10.36040/jati.v7i2.6842.
- [15] F. Bei and S. Sudin, “Analisis Sentimen Aplikasi Tiket Online Di Play Store Menggunakan Metode Support Vector Machine (Svm),” *Sismatik*, vol. 01, no. 01, pp. 91–97, 2021.
- [16] T. D. Putra, E. Utami, and M. P. Kurniawan, “Analisis Sentimen Pemilu 2024 dengan Naive Bayes Berbasis Particle Swarm Optimization (PSO),” *Explore*, vol. 13, no. 1, pp. 1–5, 2023, doi: 10.35200/ex.v11i2.13.
- [17] H. Sujadi, “Analisis Sentimen Pengguna Media Sosial Twitter Terhadap Wabah Covid-19 Dengan Metode Naive Bayes Classifier Dan Support Vector Machine,” *INFOTECH journal*, vol. 8, no. 1, pp. 22–27, 2022, doi: 10.31949/infotech.v8i1.1883.
- [18] R. Talisman, “Perancangan Aplikasi Data Crawling Untuk Pencarian Buku Pada Toko Buku Online,” vol. 11, no. 4, pp. 4333–4340, 2024.
- [19] A. Y. Nugroho, U. Cendekia, and M. Indonesia, “Penerapan Teknik Analisis Data untuk Prediksi Penjualan Exploratory Data Analysis ( EDA ),” vol. 02, no. 03, pp. 922–929, 2024.
- [20] A. Agung, A. Daniswara, I. Kadek, and D. Nuryana, “Data Preprocessing Pola Pada Penilaian Mahasiswa Program Profesi Guru,” *Journal of Informatics and Computer Science*, vol. 05, pp. 97–100, 2023.
- [21] R. Peranginangin, E. J. G. Harianja, I. K. Jaya, and B. Rumahorbo, “Penerapan Algoritma Safe-Level-Smote Untuk Peningkatan Nilai G-Mean Dalam Klasifikasi Data Tidak Seimbang,” *METHOMIKA Jurnal Manajemen Informatika dan Komputerisasi Akuntansi*, vol. 4, no. 1, pp. 67–72, 2020, doi: 10.46880/jmika.vol4no1.pp67-72.