

SENASTIKA Universitas Malikussaleh

IMPLEMENTASI DETEKSI BERITA CLICKBAIT BERBAHASA INDONESIA MENGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE

Dila Puspita^{*1}, Rizal Tjuet Adek², Hafizh Al Kautsar Aidilof^{*3}

^{1,2,3,4} Teknik Informatika, Universitas Malikussaleh, Lhokseumawe, Aceh

Email: ¹dila.200170024@mhs.unimal.ac.id, ²rizal@unimal.ac.id, ³hafizh@unimal.ac.id

Abstrak

Berita *online* semakin populer seiring dengan peningkatan penggunaan internet di masyarakat. Namun, fenomena *clickbait* muncul ketika banyak produsen berita menyesatkan pembaca dengan judul yang tidak relevan atau sensasional. Judul-judul tersebut sering kali tidak mencerminkan isi konten secara akurat, sehingga mengecewakan pembaca dan merusak reputasi media. Penelitian ini bertujuan untuk membangun sebuah sistem yang mampu mendeteksi berita *clickbait* dan *non-clickbait* secara otomatis. Sistem ini diharapkan dapat membantu pembaca mendapatkan informasi yang lebih akurat dan relevan serta meningkatkan kualitas berita online secara keseluruhan. Metode yang digunakan dalam penelitian ini adalah *Support Vector Machine* (SVM). SVM dipilih karena kemampuannya dalam mengidentifikasi *hyperplane* optimal yang memisahkan dua kelompok data, serta dapat menangani masalah *linier* dan *non-linier*. Dengan menggunakan data latih terlabel, SVM akan menemukan pola dalam fitur judul dan isi konten berita untuk melakukan klasifikasi antara *clickbait* dan *non-clickbait*. Dataset yang terdiri dari 8.660 berita telah dianotasi, dengan 6.888 berita digunakan sebagai data latih dan 1.772 berita sebagai data uji. Hasil evaluasi model menunjukkan akurasi sebesar 82%, precision 81%, recall 65%, dan F1-score 72%. Kinerja ini membuktikan bahwa metode SVM efektif dalam memisahkan berita *clickbait* dari *non-clickbait*, yang pada akhirnya dapat membantu mengurangi penyebaran informasi yang menyesatkan di platform berita *online*.

Keywords: *Berita Online, Clickbait, Non-clickbait, Klasifikasi, Support Vector Machine (SVM).*

1. PENDAHULUAN

Berita *online* semakin populer sebagai cara untuk mendapatkan informasi seiring dengan meluasnya penggunaan internet di masyarakat. Berita dapat dengan mudah diakses dan didistribusikan ke berbagai perangkat melalui internet. Namun banyak produsen berita menggunakan ini untuk menyesatkan pembaca. Fenomena menyesatkan pembaca ini terlihat antara judul berita dan konten yang disampaikan tidak relevan. Fenomena ini disebut *clickbait* [1].

Clickbait adalah jenis berita yang menggunakan judul yang sensasional atau menyesatkan untuk menarik perhatian pembaca dan meningkatkan klik. Meskipun tidak selalu mengandung informasi yang salah, *clickbait* sering kali mengecewakan pembaca karena isi kontennya yang tidak sebanding dengan ekspektasi yang dibangun oleh judulnya. Ini tidak hanya merugikan pembaca, tetapi juga dapat merusak reputasi media dan menurunkan kualitas informasi yang beredar di masyarakat. Ciri-ciri berita *clickbait* ditandai dengan penggunaan judul yang emosional, berlebihan, atau menyesatkan. Judul-judul tersebut sering kali dimaksudkan untuk menarik perhatian pembaca, terlepas dari keakuratan atau kelengkapan informasi yang disampaikan [2].

Didasarkan pada permasalahan *clickbait*, salah satu metode yang dapat digunakan dalam penelitian ini adalah menggunakan metode *Support Vector Machine*. SVM sebuah teknik yang berupaya mengidentifikasi *hyperplane* optimal yang membagi ruang input menjadi dua kelompok. SVM memiliki kemampuan untuk menangani masalah linier dan nonlinier [3]. SVM bekerja dengan mengidentifikasi *hyperplane* optimal yang membagi ruang input menjadi dua kelompok, yaitu *clickbait* dan *non-clickbait*. Dalam proses deteksi *clickbait*, SVM akan memisahkan judul dan isi konten berita yang bersifat *clickbait* dan *non-clickbait* didasarkan fitur-fitur yang disediakan. Dengan menggunakan data latih terlabel, SVM akan mengidentifikasi pola tersembunyi dalam fitur dan melakukan klasifikasi untuk setiap berita baru. Dengan pendekatan ini, diharapkan SVM dapat memberikan keakuratan yang tinggi dalam membedakan antara berita *clickbait* dan *non-clickbait*, sehingga membantu mengurangi penyebaran konten *clickbait* yang menyesatkan dan meningkatkan kualitas informasi yang disajikan dalam berita *online*.

Berdasarkan uraian di atas, penelitian ini bertujuan untuk membangun sebuah sistem yang dapat mendeteksi berita *clickbait*. Sistem ini diharapkan dapat secara efektif mengurangi penyebaran berita *clickbait* dan meningkatkan kualitas informasi yang disajikan dalam berita *online*.

2. METODE PENELITIAN

Tahapan-tahapan dari penelitian ini yang akan dijelaskan sebagai berikut:

1. *Requitmen analisis*
Pada proses pengumpulan data peneliti mendapatkan datanya dari penelitian CLIK.ID.
2. *Design*
Design atau perancangan sistem berperan penting karena disinilah dilakukannya representasi untuk sistem yang akan dibuat kedalam sebuah alur kerja sistem atau skema sistem sehingga dengan adanya perancangan sistem akan lebih terarah dalam proses pengembangan sistem nantinya.
3. *Development*
Tahapan ini yaitu mengimplentasikan metode yang dipilih ke dalam sistem memakai *source code*. Ini melibatkan penulisan, penyesuaian, dan pengujian kode program untuk memastikan efektivitas dalam konteks aplikasi.
4. *Testing*
Pengujian sistem dilaksanakan dengan maksud untuk memverifikasi apakah sistem yang dikembangkan telah beroperasi secara efektif dan sesuai dengan kebutuhan atau harapan dalam menjalankan fungsinya.
5. *Maintenance*
Tahap *maintenance* merupakan tahap akhir dari perancangan yang dibuat. *Developer* akan terus memperbaiki serta memelihara dari hasil evaluasi pada tahap sebelumnya.

2.1 Berita

berita adalah fakta dan gagasan terkini, akurat, menarik, dan dikenal luas merupakan bahan pembentuk berita. Publikasi termasuk surat kabar, radio, televisi, dan internet semuanya berperan dalam menyebarkan berita. Outlet berita hadir untuk mendidik masyarakat tentang peristiwa penting dan praktis [4].

2.2 Clickbait

Clickbait adalah judul atau tautan yang dirancang untuk menarik perhatian pembaca dengan cara yang provokatif atau menyesatkan, dengan tujuan utama untuk meningkatkan jumlah klik dan lalu lintas ke suatu situs web. Ciri-ciri berita clickbait ditandai dengan penggunaan judul yang emosional, berlebihan, atau menyesatkan. Judul-judul tersebut sering kali dimaksudkan untuk menarik perhatian pembaca, terlepas dari keakuratan atau kelengkapan informasi yang disampaikan [5].

2.3 Data Mining

Data mining adalah proses analisis data yang melibatkan penemuan pola-pola yang berharga atau berarti dalam kumpulan data yang besar, biasanya disebut sebagai database, untuk kemudian digunakan dalam pengambilan keputusan dan prediksi di masa depan. Pola-pola ini diidentifikasi dan dianalisis secara intuitif dengan menggunakan perangkat khusus [6].

2.4 Text Mining

Text mining adalah jenis data mining khusus yang bertujuan untuk menemukan pola menarik dalam sejumlah besar data berbentuk teks. Selain untuk klasifikasi, *text mining* juga berguna untuk menangani tugas-tugas seperti ekstraksi informasi (informasi ekstraksi), pengelompokan data (pengelompokan data), dan pencarian informasi (informasi pencarian) [7].

2.5 Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah metode yang menghasilkan prediksi yang solid. Klasifikasi teks yang memanfaatkan bobot indeks kata sebagai fitur merupakan tujuan penggunaan metode SVM. Pada awal tahun 1992, Boser, Guyon, dan Vapnik membawa pembelajaran SVM ke khalayak yang lebih luas [8]. SVM merupakan suatu teknik yang mampu menyelesaikan permasalahan baik linier maupun non linier. Dalam ruang kerja berdimensi tinggi, ide kernel dipakai untuk menemukan *hyperplane* yang memaksimalkan margin antar kelas data, sehingga membantu mengatasi kesulitan non-linier. Kelas +1 dan kelas -1 merupakan dua kelompok dengan pola yang berbeda. *hyperplane* adalah alat yang hebat untuk membedakannya. Persamaan 1 sampai 2 merupakan persamaan dari *support vector machine linier*.

Persamaan 1 Menurut Vapnik dan Cortes (1995) dapat diperoleh:

$$X_i.W + b \geq 1 \text{ untuk } Y_i = 1 \quad (1)$$

$$X_i.W + b \geq -1 \text{ untuk } Y_i = -1 \quad (2)$$

Persamaan 2 mencari *hyperplane* digunakan untuk memaksimalkan jarak antara kelas data memakai persamaan berikut:

$$f(x) = w.x + b \quad (3)$$

Sedangkan ada beberapa fungsi non-linier seperti, linier, polinomial, radial basic funtion dan sigmoid . Dimana rumus sebagai berikut:

1. *Linier kernel*

$$K(x_p, x_d) = X_i^T X_j \quad (4)$$

2. *Polynomial*

$$K(x_p, x_d) = (X_i^T X_j + 1)^d, \gamma > 0 \quad (5)$$

3. *Radial Basic Funsion (RBF)*

$$K(x_p, x_d) = \exp(-\gamma \|X_i - X_j\|^2), \gamma > 0 \quad (6)$$

4. *Sigmoid*

$$K(x_p, x_d) = \tan \gamma X_i^T X_j + r \quad (7)$$

2.6 Term Frecuency-Invers Document Frecuency (TF-IDF)

Salah satu metode analisis teks yang digunakan untuk menentukan seberapa penting setiap kata dalam sebuah dokumen. Untuk melakukan ini, maka perlu menentukan frekuensi dokumen invers dan frekuensi term. Sementara TF menghitung jumlah kali kata muncul dalam sebuah dokumen, IDF menghitung berapa kali kata tersebut muncul secara unik di dalam korpus dokumen. Ketika kedua nilai ini digabungkan, TF-IDF menyorot kata-kata yang sering muncul dalam satu teks namun jarang muncul di dokumen lain selama pengambilan informasi dengan memberinya bobot lebih tinggi. Adapun beberapa tahapan dalam pembobotan *Term Frecuency-Inverse Document Frecuency* (TF-IDF) adalah:

1. Hitung *term frequency*(TF)

Term Frecuency ialah frekuensi kemunculan data dalam dokumen teks . *term frecuency*($\mathbf{tf}_{t,d}$) didefenisikan bayaknya term t muncul dalam dokumen.

$$W_{\mathbf{tf}_{t,d}} = \left\{ \begin{array}{l} 1 + \log 10 \mathbf{tf}_{t,d}, \text{ if } \mathbf{tf}_{t,d} > 0 \\ 0, \text{ otherwise} \end{array} \right\} \quad (8)$$

2. Hitung *bobot invers document frequency*(IDF)

Invers Dokument Frecuency yaitu hubungan kemunculan term diseluruh dokumen teks.

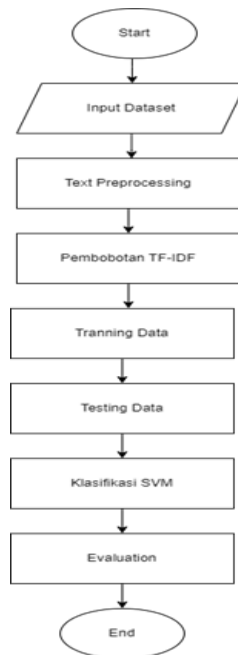
$$idf_t = \log \frac{N}{df_t} \quad (9)$$

3. Hitung *nilai bobot Term Frecuency-Invers Document Frecuency TF-IDF*

Nilai tf-idf dari suatu kata merupakan gabungan dari nilai tf dan nilai idf dalam perhitungan bobot.

$$W_{t,d} = W_{\mathbf{tf}_{t,d}} \times idf_t \quad (10)$$

3. Skema Sistem



Gambar 1 skema sistem

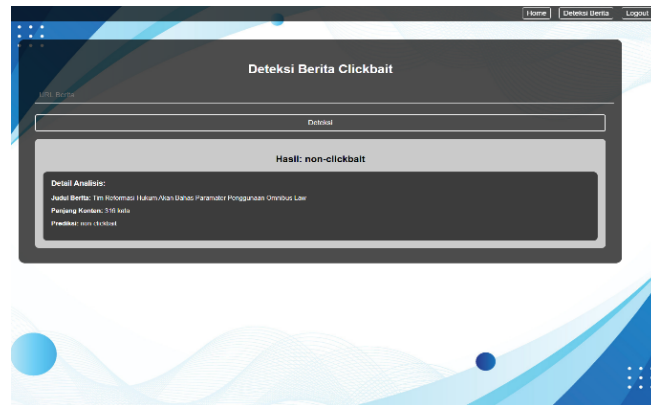
Sistem dimulai dengan mengidentifikasi masalah klasifikasi berita, yang melibatkan input dataset berita yang sudah dilabeli, dilanjutkan dengan langkah *text preprocessing* untuk membersihkan dan normalisasi teks, diikuti oleh pembobotan TF-IDF untuk mewakili teks dalam bentuk vektor. Langkah selanjutnya adalah training data dan testing data sehingga model klasifikasi SVM dapat dilatih pada separuh kumpulan data dan diuji pada separuh lainnya. Representasi vektor TF-IDF dipakai oleh SVM untuk klasifikasi kategori berita. Akurasi, presisi, perolehan, dan skor F1 adalah beberapa ukuran yang dipakai dalam evaluasi model. Pada akhirnya, sistem akan melakukan evaluasi model dengan cermat, dan model yang dilatih dapat dimanfaatkan dengan baik dengan mengategorikan berita baru.

4. HASIL DAN PEMBAHASAN

Penelitian ini juga mengembangkan sebuah sistem yang dapat mendeteksi berita clickbait dan non-clickbait dengan menerapkan metode *Support Vector Machine* (SVM). Metode SVM dipilih karena kemampuannya untuk mengklasifikasikan data secara efisien. Sistem ini bekerja dengan menerima input berupa URL dari berita yang ingin diperiksa, kemudian mengklasifikasikan berita tersebut ke dalam kategori *clickbait* atau *non-clickbait*. Model yang dikembangkan menggunakan SVM menunjukkan kinerja yang baik dengan akurasi yang memadai, sehingga sistem ini mampu memberikan hasil klasifikasi yang akurat dan cepat.

3.1 Hasil Implementasi Sistem

Tahapan implementasi sistem merupakan serangkaian langkah yang harus dilalui dalam proses pengembangan perangkat lunak yang telah dirancang, dianalisis, dan dibangun. Setelah proses pengembangan selesai, sistem akan melalui tahap pengujian untuk memastikan bahwa sistem tersebut memenuhi standar kelayakan dan berfungsi sesuai dengan spesifikasi serta tujuan yang telah ditetapkan. Hasil pengujian sistem dapat dilihat pada gambar dibawah ini.



Gambar 2 Pengujian sistem

Pada pengujian di atas, pengguna dapat dengan mudah memasukkan URL berita yang ingin dianalisis ke dalam sistem. Setelah URL dimasukkan, sistem secara otomatis mengambil judul dan konten dari artikel tersebut. Proses ini berlangsung cepat, sehingga pengguna tidak perlu melakukannya secara manual. Setelah data diambil, sistem akan menganalisis konten dan memberikan hasil klasifikasi yang jelas, menunjukkan apakah berita tersebut termasuk dalam kategori *clickbait* atau *non-clickbait*.

3.2 Pengumpulan Data

Penelitian ini menggunakan dataset berita berbahasa Indonesia yang diambil dari penelitian sebelumnya atau sumber CLIK.ID. data yang digunakan dalam penelitian ini yaitu judul dan konten berita yang telah diberi label berdasarkan kategori *clickbait* atau *non-clickbait*. Data yang digunakan dalam penelitian ini berjumlah 8.660 berita yang sudah dianotasi. data berita yang diperoleh dari CLICK.ID dapat dilihat pada gambar di bawah ini.

	judul	konten	label
0	Masuk Radar Pilwalkot Medan, Menantu Jokowi Be...	Medan - Menantu Jokowi, Bobby Afif Nasution, m...	non-clickbait
1	Malaysia Sudutkan RI: Isu Kabut Asap hingga In...	Jakarta - Malaysia kembali menyerang pemerinta...	non-clickbait
2	Viral! Driver Ojol di Bekasi Antar Pesanan Mak...	Bekasi - Seorang driver ojek online di Bekasi ...	clickbait
3	Kemensos Salurkan Rp 7,3 M bagi Korban Kerusuh...	Jakarta - Kementerian Sosial (Kemensos) menyal...	non-clickbait
4	MPR: Amandemen UUD 1945 Tak Akan Melebar ke Ma...	Jakarta - Anggota Badan Pengkajian MPR RI Bamb...	non-clickbait

Gambar 3 Data Berita

3.3 Preprocessing

Untuk membangun aplikasi deteksi berita *clickbait* dan *non-clickbait*, terdapat beberapa langkah penting yang dilakukan, terutama pada tahap *preprocessing*. *Preprocessing* pada *text mining* sangat penting dilakukan untuk meningkatkan kualitas data yang digunakan oleh model. Pada penelitian ini menggunakan teknik *preprocessing* yang dilakukan meliputi tokenisasi, di mana teks dipecah menjadi kata-kata atau token yang lebih kecil, penghilangan stopwords atau kata-kata yang tidak memiliki makna signifikan, serta stemming untuk mengubah kata-kata menjadi bentuk dasar. Berikut merupakan data latih berita yang sudah di tokenisasi, stopwords dan stemming dapat dilihat pada tabel di bawah ini.

Tabel 1 Tokenisasi, Stopword Dan Steaming

Preprocessing	Judul	konten	label
<i>Tokenisasi</i>	viral driver ojol di bekasi antar pesanan makanan pakai sepeda	seorang driver ojek online di bekasi eko susilo menjadi viral di media sosial karena mengantarkan pesanan makanan menggunakan sepeda	<i>Clickbait</i>
<i>Stopword</i>	viral driver ojol bekasi pesanan makanan pakai sepeda	seorang driver ojek online bekasi eko susilo viral media sosial mengantarkan pesanan makanan sepeda	<i>Clickbait</i>

Steaming viral driver ojol bekas pesan seorang driver ojek online bekas eko susilo
makan pakai sepeda viral media sosial antar pesan makan sepeda *Clickbait*

3.4 Term Frequency-Inverse Document Frequency TF-IDF

Pada penelitian ini, dilakukan pembobotan kata yang merupakan tahap penting dalam pemrosesan teks. Tahap pembobotan ini bertujuan untuk memberikan nilai bobot pada setiap kata dalam dokumen dengan menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF). TF-IDF adalah teknik yang menghitung seberapa sering suatu kata muncul dalam sebuah dokumen (*Term Frequency*) dan seberapa jarang kata tersebut muncul di seluruh kumpulan dokumen (*Inverse Document Frequency*). Bobot yang dihasilkan oleh TF-IDF membantu dalam menentukan pentingnya sebuah kata dalam konteks tertentu, sehingga kata-kata yang lebih relevan akan memiliki bobot lebih tinggi, sementara kata-kata umum yang muncul di banyak dokumen akan diberi bobot yang lebih rendah. Dengan demikian, pembobotan kata menggunakan TF-IDF membantu meningkatkan akurasi model dalam mengenali pola pada teks, terutama untuk analisis seperti deteksi *clickbait*. Tabel dibawah ini merupakan Langkah dalam menghitung proses TF-IDF.

Tabel 2. Proses Perhitungan TF-IDF

TERM	TF	IDF	TF-IDF
viral	4	0,30103	1,20412
driver	3	0,30103	0,90309
ojol	3	0,30103	0,90309
bekas	5	0,30103	1,50515
pesan	4	0,30103	1,20412
makan	3	0,30103	0,90309
pakai	2	0,30103	0,60206
sepeda	5	0,30103	1,50515
seorang	1	0,30103	0,30103
ojek	1	0,30103	0,30103
online	1	0,30103	0,30103
eko	8	0,30103	0,30103
susilo	1	0,30103	2,40824
media	1	0,30103	0,30103
.....
sosial	1	0,30103	0,30103

3.5 Pembagian Data

Langkah ini bertujuan untuk membagi data menjadi data latih dan data uji. Pembagian ini bertujuan untuk melatih model deteksi *clickbait* menggunakan algoritma *machine learning* serta mengukur kinerja model berdasarkan hasil pengujian terhadap data uji.

Tabel 3. Pembagian Data

Keseluruhan Data	Data Latih	Data Uji
8.660	80% 6.888	20% 1.722

3.6 Pelatihan Model SVM

Implementasi *Support Vector Machine* (SVM) dalam penelitian ini menggunakan bahasa pemrograman python. Python memiliki *library Sklearn* yang dapat digunakan untuk mengimplementasikan SVM. Baris code implementasi SVM dapat dilihat pada gambar.

```

# Train SVM with linear kernel
model =SVC(C=1.0, kernel='linear', gamma='auto')
model.fit(X_train, y_train)

# Save model and vectorizer
joblib.dump(model, 'models/svm_clickbait_model.pkl')
joblib.dump(vectorizer, 'models/vectorizer.pkl')

# Make predictions on test data
y_pred = model.predict(X_test)

```

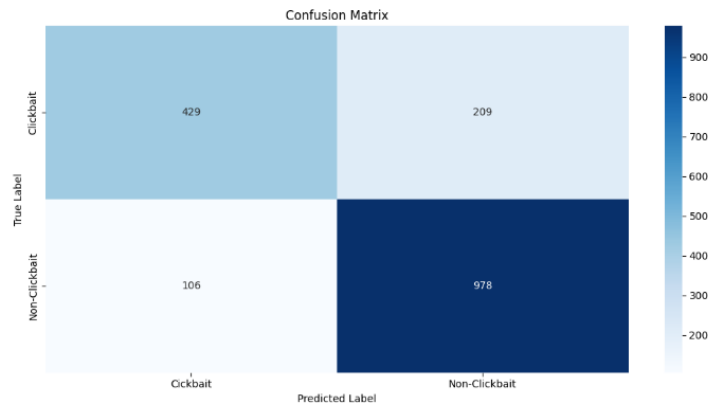
Gambar 4 Pemodelan SVM

Kode di atas merupakan *Model Support Vector Classifier (SVC)* dengan kernel linear digunakan untuk mengklasifikasikan berita antara clickbait dan non-clickbait. Dengan parameter $C=1.0$ dan $\text{gamma}='auto'$, model dilatih menggunakan fitur TF-IDF dari teks berita. Setelah pelatihan, model dan vectorizer disimpan dalam file untuk penggunaan ulang. Model diuji dengan data uji, dan hasil evaluasi dilakukan menggunakan confusion matrix untuk menilai performa prediksi.

3.7 Evaluasion

Evaluasi ini dilakukan untuk mengetahui tingkat keberhasilan kinerja model dalam melakukan proses klasifikasi *Clickbait* dan *Non-Clickbait*. dari *confusion matrix* dapat diketahui akurasi, presisi dan recall. Pada penelitian ini menggunakan 2 kelas sehingga parameter perhitungan *confusion matrix* terdiri dari 4 hal yaitu *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, *False Negative (FN)*.

Data yang digunakan dalam pelatihan terdiri dari 6.888 berita, sedangkan data pengujian berjumlah 1.772 berita. Berdasarkan tabel *confusion matrix*, hasil prediksi menunjukkan bahwa:



Gambar 5 Confusion matrix

Dari *confusion matrix* diatas kita bisa menghitung beberapa metrik evaluasi penting:

```

Classification Report:
              precision    recall  f1-score   support

   clickbait         0.80      0.67      0.73         638
  non-clickbait         0.82      0.90      0.86        1084

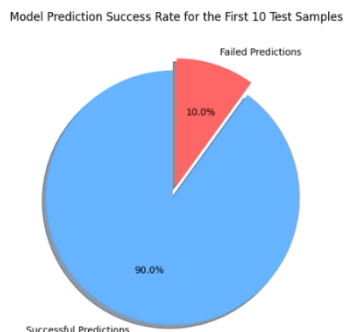
   accuracy                   0.82         1722
  macro avg              0.81      0.79      0.80         1722
 weighted avg              0.82      0.82      0.81         1722

```

Gambar 6 Klasifikasi report

Berdasarkan hasil pengujian di atas dengan metode SVM dalam mendeteksi *clickbait* dengan menggunakan kernel linier mendapatkan akurasi sebesar *accuracy* 82%, *precision* 81%, *recall* 65%, dan *F1-score* 72%.

3.8



**Grafik
Keberhasilan
Model**

Gambar 7 Tingkat Keberhasilan model

Grafik ini menunjukkan bahwa model memiliki tingkat keberhasilan sebesar 90% dan tingkat kegagalan hanya 10% untuk 10 data uji pertama. Ini mengindikasikan bahwa model mampu memprediksi dengan baik antara *clickbait* dan *non-clickbait*.

4. DISKUSI

Hasil penelitian ini menunjukkan bahwa metode *Support Vector Machine* (SVM) yang diterapkan untuk mendeteksi berita *clickbait* dan *non-clickbait* mampu memberikan kinerja yang baik. Model ini mencapai akurasi sebesar 82%, *precision* 81%, *recall* 65%, dan *F1-score* 72%. Selain itu, grafik yang dihasilkan memperlihatkan tingkat keberhasilan model mencapai 90%, dengan tingkat kegagalan hanya 10% untuk 10 data uji pertama. Ini mengindikasikan bahwa model memiliki kemampuan yang baik dalam membedakan antara *clickbait* dan *non-clickbait*. Kinerja ini menggarisbawahi efektivitas SVM dalam mengklasifikasikan teks berita berdasarkan karakteristik *clickbait*.

Penelitian yang dilakukan oleh Wanda Athira Luqyana berjudul Analisis Sentimen Cyberbullying pada Komentar Instagram dengan metode klasifikasi Support Vector Machine [3], menunjukkan hasil klasifikasi yang membedakan antara kelas positif dan negatif, dengan fokus pada sentimen positif dan negatif terkait cyberbullying. Berdasarkan pengujian, tingkat akurasi tertinggi yang dicapai adalah 90%, disertai *precision* 94,44%, *recall* 85%, dan *F-measure* 89,47%. Penelitian ini menggunakan komposisi data latih 50% dan data uji 50%.

Jika dibandingkan, penelitian ini mengenai klasifikasi berita *clickbait* menggunakan model *Support Vector Classifier* (SVC) memperoleh akurasi sebesar 82%, *precision* 81%, *recall* 65%, dan *F1-score* 72%. Meskipun hasil akurasi dan *precision* menunjukkan performa yang baik, nilai *recall* yang lebih rendah mengindikasikan bahwa model belum sepenuhnya optimal dalam mendeteksi semua berita *clickbait*. Hal ini mungkin disebabkan oleh ketidakseimbangan data antara kelas *clickbait* dan *non-clickbait*, di mana jumlah berita *clickbait* jauh lebih sedikit.

5. KESIMPULAN

Dari hasil pengujian yang telah dilakukan, dapat ditarik beberapa kesimpulan dalam mendeteksi berita *clickbait* pada yang menerapkan metode SVM. Hal yang disimpulkan dari penelitian ini adalah sebagai berikut:

1. Penelitian ini mengembangkan sistem yang dapat deteksi berita *clickbait* dan *non-clickbait* dengan menerapkan metode *Support Vector Machine* (SVM). Sistem ini berfungsi dengan menerima input berupa URL dari berita yang ingin diperiksa, lalu mengklasifikasikan berita tersebut kedalam kategori *clickbait* dan *non-clickbait*. Model yang dikembangkan menggunakan metode SVM menunjukkan kinerja yang baik dengan akurasi yang memadai.
2. Model *Support Vector Classifier* (SVC) dengan kernel linear digunakan untuk mengklasifikasikan berita antara *clickbait* dan *non-clickbait*. Hasil pengujian menunjukkan akurasi sebesar 82%, *precision* 81%, *recall* 65%, dan *F1-score* 72%. Selain itu, grafik menunjukkan bahwa model memiliki tingkat keberhasilan sebesar

90% dan tingkat kegagalan hanya 10% untuk 10 data uji pertama. Ini mengindikasikan bahwa model mampu memprediksi dengan baik antara *clickbait* dan *non-clickbait*.

6. UCAPAN TERIMA KASIH

Terima kasih yang sebesar-besarnya saya sampaikan kepada dosen pembimbing yang telah memberikan bimbingan, serta kepada orang tua dan keluarga yang telah memberikan dukungan dan bantuan tanpa henti selama proses penelitian ini. Dukungan moril dan materiil yang diberikan sangat berarti, sehingga penelitian ini dapat berjalan dengan lancar dan mencapai hasil yang lebih baik. Ucapan terima kasih juga saya sampaikan kepada Universitas Malikussaleh yang telah memberikan fasilitas untuk menunjang penelitian ini. Semoga segala kebaikan dan bantuan yang diberikan mendapatkan balasan yang setimpal.

7. DAFTAR PUSTAKA

- [1] W. Afandi, S. N. Saputro, A. M. Kusumaningrum, H. Adriansyah, M. H. Kafabi, and S. Sudioanto, "Klasifikasi Judul Berita Clickbait menggunakan RNN-LSTM," *J. Inform. J. Pengemb. IT*, vol. 7, no. 2, pp. 85–89, 2022, doi: 10.30591/jpit.v7i2.3401.
- [2] A. F. Yavi, "Klasifikasi Artikel Berbahasa Indonesia untuk Mendeteksi Clickbait menggunakan Metode Naïve Bayes," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2018.
- [3] W. Athira Luqyana, I. Cholissodin, and R. S. Perdana, "Analisis Sentimen Cyberbullying pada Komentar Instagram dengan Metode Klasifikasi Support Vector Machine," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 11, pp. 4704–4713, 2018, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [4] E. P. Bangun, F. V. I. A Koagouw, and J. S. Kalangi, "Analisis Isi Unsur Kelengkapan Berita Pada Media Online Manadopostonline.com," *Acta Diurna Komun.*, vol. 1, no. 3, pp. 4–13, 2019, [Online]. Available: <https://ejournal.unsrat.ac.id/index.php/actadiurnakomunikasi/article/view/25560>
- [5] R. Sagita, U. Enri, and A. Primajaya, "Klasifikasi Berita Clickbait Menggunakan K-Nearest Neighbor (KNN)," *JOINS (Journal Inf. Syst.)*, vol. 5, no. 2, pp. 230–239, 2020, doi: 10.33633/joins.v5i2.3705.
- [6] A. Deolika, K. Kusriani, and E. T. Luthfi, "Analisis Pembobotan Kata Pada Klasifikasi Text Mining," *J. Teknol. Inf.*, vol. 3, no. 2, p. 179, 2019, doi: 10.36294/jurti.v3i2.1077.
- [7] R. Tjut, Adek, H. Al Kautsar Aidilof, and W. Kurniawan, "Klasifikasi Berita Olahraga Pada Portal Berita Online Dengan Metode K-Nearest Neighbour (Knn) Dan Levenshtein Distance," *J. Teknol. Terap. Sains 4.0*, vol. 2, no. 1, p. 365, 2021, doi: 10.29103/tts.v2i1.3760.
- [8] I. P. Monika and M. T. Furqon, "Penerapan Metode Support Vector Machine (SVM) Pada Klasifikasi Penyimpangan Tumbuh Kembang Anak," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 10, pp. 3165–3166, 2018, [Online]. Available: <http://j-ptiik.ub.ac.id>