IMPLEMENTASI ALGORITMA *CONVOLUTIONAL NEURAL NETWORK* UNTUK MENGIDENTIFIKASI BERITA HOAKS BERBAHASA INDONESIA

Vito Ramadhan, Asriyanik, Agung Pambudi
Teknik Informatika, Universitas Muhammadiyah Sukabumi
Jl. R. Syamsudin, S.H. No. 50 43113 Sukabumi Jawa Barat · 76 km
vitoramadhan802@ummi.ac.id

ABSTRAK

Penyebaran berita hoaks, khususnya dalam sektor politik, telah menjadi masalah serius di era digital yang dapat menimbulkan kebingungan dan konflik sosial. Penelitian ini bertujuan untuk mengembangkan model klasifikasi berita hoaks berbahasa Indonesia di Facebook dengan menggunakan pendekatan CRISP-DM dan model *Convolutional Neural Network* (CNN). Proses dimulai dengan pemahaman bisnis terkait penyebaran hoaks, dilanjutkan dengan pembersihan teks, tokenisasi, dan lemmatization data. Data kemudian dibagi menjadi data latih dan uji untuk pengembangan model CNN. Hasil penelitian menunjukkan model mencapai akurasi 92,53% pada data pelatihan dan 81,09% pada data pengujian, dengan *loss* 0,33 dan 0,55. Evaluasi menggunakan *confusion matrix* serta metrik *precision*, *recall*, dan F1-*score* menunjukkan model ini efektif dalam mendeteksi berita hoaks politik dan dapat digunakan untuk meningkatkan keakuratan identifikasi konten hoaks.

Kata kunci : *Berita, Hoaks, Convolutional Neural Network (CNN), Confusion matrix.*

1. PENDAHULUAN

Berita palsu atau hoaks merupakan masalah serius di era digital saat ini, di mana informasi yang tidak benar sengaja dipublikasikan untuk menipu masyarakat. Hoaks sering kali digunakan dengan tujuan jahat, untuk memanipulasi atau menyesatkan pembaca agar percaya pada informasi yang sebenarnya tidak benar. Di Indonesia, tingkat literasi media yang rendah memperburuk masalah ini, karena banyak orang tidak memeriksa kebenaran berita sebelum membagikannya. Hal ini dapat menyebabkan penyebaran informasi yang salah dengan cepat, yang dapat menimbulkan kebingungan, dan konflik sosial yang merusak persatuan masyarakat [1].

Menurut data dari Kementerian Komunikasi dan Informatika, pada tahun 2023 telah ditangani 1.615 konten hoaks di media sosial, dengan total 12.547 konten sejak 2018. Sebagian besar hoaks berkaitan dengan sektor kesehatan, terutama terkait dengan Covid-19, diikuti oleh hoaks tentang kebijakan pemerintah dan penipuan. Hoaks politik juga menjadi masalah signifikan, dengan 1.628 kasus terkait partai politik dan pemilu sejak 2018. Penyebaran hoaks ini berdampak pada masyarakat secara langsung, dengan merusak kepercayaan publik, mempengaruhi opini masyarakat, dan mengubah keputusan politik. Ketidakpastian dan kebingungan yang ditimbulkan dari hoaks dapat menyebabkan ketidakstabilan sosial dan politik secara lebih luas [2].

Dampak penyebaran berita hoaks bisa sangat merugikan, dapat mencakup aspek ekonomi, hubungan sosial, dan psikologis. Kerugian materiil dan non-materiil, dampak psikologis, dan hilangnya kepercayaan masyarakat adalah beberapa konsekuensi dari penyebaran informasi yang salah. Untuk mengatasi masalah ini, penting untuk mengidentifikasi hoaks dengan metode yang efektif. Seperti Klasifikasi teks dengan menggunakan teknik pembelajaran mendalam (Deep Learning) dapat menjadi solusi yang efisien dalam mengidentifikasi dan menangani berita hoaks secara cepat dan tepat [2].

Metode *Convolutional Neural Network* (CNN) merupakan salah satu pendekatan yang menjanjikan dalam pengidentifikasian berita hoaks. CNN karena dapat memproses data teks dengan baik, dengan mengidentifikasi pola-pola lokal dan fitur penting secara efisien melalui lapisan konvolusi dan max pooling. dengan mengeksplorasi arsitektur CNN serta metode penggabungan fitur, dengan harapan dapat mengembangkan dan meningkatkan akurasi model dalam mendeteksi berita hoaks dan mengurangi dampak negatif dari penyebaran informasi yang salah secara tepat[3].

2. TINJAUAN PUSTAKA

2.1. Hoaks

Hoaks adalah informasi yang sengaja dimanipulasi untuk menutupi kebenaran. Secara sederhana, hoaks adalah usaha memutarbalikkan fakta dengan informasi yang terlihat meyakinkan namun tidak benar. Tujuan utama hoaks adalah menciptakan ketidakamanan, ketidaknyamanan, dan kebingungan di masyarakat. Dalam kebingungan, masyarakat cenderung membuat keputusan yang lemah, tidak meyakinkan, atau bahkan salah[4].

2.2. *Deep Learning*

Deep Learning adalah cabang dari *Machine Learning* yang terinspirasi oleh struktur otak manusia. Struktur ini dikenal sebagai *Artificial Neural Networks* (ANN) atau Jaringan Saraf Tiruan (JST)[5].

Proses pembelajaran *Deep Learning* bisa berupa *supervised learning*, *semi supervised learning*, atau *unsupervised learning*. *Supervised learning* menggunakan data dengan informasi yang sudah ada, *semi supervised learning* menggunakan data yang

tidak lengkap, dan *unsupervised learning* belajar tanpa masukan data. Untuk klasifikasi citra, sering digunakan Jaringan Saraf Tiruan (ANN) seperti *Convolutional Neural Network* (CNN), yang menghubungkan beberapa lapisan pemrosesan dan terinspirasi oleh sistem saraf biologis [6].

2.3. *Natural Language Processing*

Natural Language Processing (NLP) adalah cabang ilmu komputer dalam bidang kecerdasan buatan yang berfokus pada interaksi antara komputer dan bahasa alami manusia, seperti bahasa Indonesia atau bahasa Inggris, serta berkaitan erat dengan linguistic[7].

NLP adalah disiplin ilmu dalam komputer, kecerdasan buatan, dan linguistik yang fokus pada interaksi antara komputer dan bahasa alami manusia. Bahasdigunakan untuk menyampaikan informasi antarindividu, baik dalam bentuk suara maupun teks [8].

2.4. *Convolutional Neural Network*

Convolutional Neural Network (CNN) adalah metode jaringan saraf yang menggunakan kernel dua dimensi untuk konvolusi pada setiap lapisan. CNN mempelajari fitur objek melalui konvolusi dan menggabungkan fitur spasial untuk membuat prediksi, mengurangi jumlah variabel dengan menggunakan beberapa parameter[9].

Berikut adalah arsitektur dari CNN:

2.5. Ekstraksi Fitur

Lampiran Ekstraksi Fitur berfungsi untuk melakukan operasi konvolusi pada output dari layer sebelumnya. Adapun rumus dari ekstraksi fitur adalah sebagai berikut:

$$FM_{(i_l, j_l)}^{(l, m_l)} = f\left(\sum_{r_l=0}^{k_h} \sum_{c_l}^{k_w} C_{(r_l, c_l)}^{(l, m_l)} * FM_{(r_l + i_{l-1}),\ (c_l + j_l - 1)}^{(l-1)}\right) \quad (1)$$

Dengan keterangan sebagai berikut:
FM = *Feature Map*
l = Index *Layers* atau *Input*
m_l = Index map pada *layers* ke-1
i_l = Index baris *FM layers* ke-1
j_l = Index kolom FM *layers* ke-1
k_h = Tinggi Kernel
r_l = Panjang kernel *layers* ke-1
k_w = Panjang kernel
c_l = Tinggi kernel *layers* ke-1
C = Kernel Konvolusi

2.6. *Fully Connected Layer*s

Arsitektur *Multilayers* Perceptron digunakan untuk efektivitas selama proses pelatihan. Jumlah neuron pada hidden layers dapat ditentukan menggunakan persamaan 2. Fungsi *Rectified Linear* Unit (ReLU) terdapat pada persamaan 3, dan untuk mengurangi dimensi peta fitur (*Pooling*), digunakan persamaan 4.

$$\boldsymbol{p = \sqrt{(m\ x\ n)}} \quad (2)$$

Dengan keterangan:
m = Jumlah Neuron Input
n = Jumlah Neuron Output

$$\mathbf{ReLU}(x): \begin{cases} x, & if x > 0 \\ 0, & otherwise \end{cases} \quad (3)$$

Dengan keterangan:
x = input vector s

$$f(x) = max(0, x) \quad (4)$$

Dengan keterangan:
x = input vector

Arsitektur Multilayers Perceptron digunakan untuk efektivitas selama proses pelatihan. Jumlah neuron pada hidden layers dapat ditentukan menggunakan persamaan 2. Fungsi Rectified Linear Unit (ReLU) terdapat pada persamaan 3, dan untuk mengurangi dimensi peta fitur (Pooling), digunakan persamaan 4.

$$\boldsymbol{\sigma_i(a) = \frac{e^a i}{\sum_j e^a j}} \quad (5)$$

Dengan keterangan:
$\sigma_i(a)$ = Nilai *output* ke-i
$e^a i$ = Nilai unit ke-i
$\sum_j$ = Jumlah neuron lapisan *output*
$e^a j$ = Nilai neuron ke-j

2.7. Feed Forward

Feed Forward adalah tahap di mana *neuron* input $\boldsymbol{x_i}$ mengirimkan sinyal ke hidden layers, yang kemudian dikalikan dengan bobot dan ditambahkan nilai bias.

$$\boldsymbol{z\_net\ j = vj0 + \sum n\ xi\ vkj} \quad (6)$$

Dengan keterangan:
i = Indeks neuron *input*
j = Indeks neuron *hidden*
$vj0$ = Bobot bias dari *input layers* *hidden layers*
n = Jumlah neuron pada *input layers*
xi = Data nilai masukan yang ke-i
vkj = Bobot dari *input layers* ke *hidden layers*

Kemudian, setiap neuron output menghitung dengan mengalikan nilai bobot dan menambahkan nilai bias.

$$\boldsymbol{y_{net}k = wk0 + \sum p\ zj\ ykj} \quad (7)$$

Dengan keterangan:
j = Indeks neuron *hidden layers*
p = Jumlah neuron pada *hidden layers*
k = Indeks neuron *output*
$wk0$ = Bobot bias dari *hidden layers* ke *output layers*
zj = Nilai hasil aktivasi dari *hidden layers*
ykj = Bobot dari *hidden layers* ke *output layers*

2.8. Back Propagation

Back propagation atau fase mundur adalah tahap di mana neuron output menerima pola target sesuai dengan pola input selama pelatihan dan menghitung perubahan nilai bias.

$$\Delta \boldsymbol{wk} = \boldsymbol{\alpha \, x \, \delta k} \quad (8)$$

Dengan keterangan:
$\boldsymbol{\alpha}$ = Nilai *learning rate*
$\boldsymbol{\delta k}$ = Komponen kesalahan pada *output layers*

2.9. *CRISP-DM*

CRISP-DM adalah metodologi standar dalam data mining yang menyediakan kerangka kerja untuk menjalankan tugas-tugas data mining. Metodologi ini memberikan proses standar untuk menangani masalah bisnis melalui data mining, dengan keunggulan mudah diimplementasikan karena setiap tahap dijelaskan secara terperinci, terstruktur, dan terdokumentasi dengan baik [10].

a. *Bussines Understanding*
Tahap ini bertujuan untuk memahami tujuan bisnis, situasi, dan menentukan sasaran penelitian.
b. *Data Understanding*
Tahap ini bertujuan untuk mengumpulkan dan mendeskripsikan data, mengeksplorasi, dan mengidentifikasi masalah kualitas data serta subset data awal.
c. *Data Preparation*
Tahap ini bertujuan untuk mempersiapkan data melalui *Cleaning*, *Case Folding*, *Tokenizing*, *Lemmatization*.
- *Cleaning*
- *Case Folding*
- *Tokenizing*
- *StopWord*s
- *Lemmatization*

d. *Modelling*
Tahap ini bertujuan untuk menentukan teknik data mining dan mencari hubungan dalam data sesuai tujuan.
e. *Evaluation*
Tahap ini bertujuan untuk menilai apakah model memenuhi kriteria dan menjalankan semua langkah dengan benar.
f. *Deployment*
Tahap ini bertujuan untuk mengembangkan dan mendokumentasikan rencana penyebaran, pemantauan, dan pemeliharaan model, serta menghasilkan laporan akhir proyek.

2.10. *Confusion matrix*

Confusion matrix adalah tabel dengan 4 kombinasi nilai prediksi dan aktual yang menunjukkan jumlah data uji yang benar dan salah. Tabel ini digunakan untuk menentukan persentase akurasi, presisi, dan *recall* [11].

Ada 4 istilah utama dalam *confusion matrix*, yaitu:

Tabel 1. Istilah *Confusion matrix*

True Positive (TP)	Data positif yang terdeteksi benar.
True Negative (TN)	Data negatif yang terdeteksi benar.
False Positive (FP)	Data negatif yang terdeteksi sebagai positif
False Negative (FN)	Data positif yang terdeteksi sebagai negatif.

3. METODE PENELITIAN

3.1. Metode Penelitian

Sebagai metode penelitian untuk studi ini, akan diterapkan metodologi CRISP-DM. Penggunaan metode ini bertujuan untuk memastikan penelitian dilakukan secara terstruktur dan sistematis. Tahapan dari metode ini adalah sebagai berikut:

3.2. *Business Understanding*

Tahap awal penelitian ini adalah business understanding, bertujuan menentukan tujuan dalam mengidentifikasi berita hoaks. Proses ini mengidentifikasi masalah penyebaran hoaks politik yang merugikan masyarakat. Penelitian ini akan mengembangkan model *Convolutional Neural Network* (CNN) untuk mengidentifikasi berita hoaks dengan akurasi tinggi, memungkinkan penanganan hoaks politik secara lebih efisien.

a. *Data Understanding*
Tahap selanjutnya adalah data understanding, di mana dilakukan pengumpulan dan eksplorasi data. Dataset yang digunakan adalah berita politik yang diperoleh melalui scraping dari *Facebook*. Dataset ini akan dibagi menjadi dua kategori valid dan hoaks.
b. *Data Preprocessing*
Pada tahap data *preprocessing*, dilakukan penyesuaian agar dataset teks berita politik siap digunakan dalam pemodelan. Proses data *preprocessing* meliputi:
- *Cleaning* Data
Menghapus elemen tidak relevan dari teks, seperti karakter khusus, tag HTML, dan spasi ganda.
- *Case Folding*
Mengubah semua huruf menjadi huruf kecil untuk mengurangi variasi teks akibat penggunaan huruf kapital.
- *Tokenizing*
Memecah teks menjadi unit-unit kecil (token) untuk membantu model menganalisis dan memahami struktur serta isi teks.

- *StopWord Removal*
  Menghilangkan kata-kata umum seperti 'dan', 'atau', yang tidak memberikan informasi penting pada proses pemodelan.
- *Lemmatization*
  Mengubah kata ke bentuk dasarnya, seperti 'berlari' dan 'pelari' menjadi 'lari', untuk mengelompokkan kata-kata yang memiliki arti sama.
- Vektorisasi Kata (Word2Vec)
  Mengubah kata-kata menjadi representasi vektor untuk menangkap makna semantik antar kata berdasarkan konteksnya.

3.3. *Modelling*

Pada tahap pemodelan, akan dikembangkan model prediktif menggunakan algoritma CNN untuk membedakan berita politik yang valid dari yang berpotensi hoaks. Alur proses pemodelan adalah sebagai berikut:

Tabel 2. Alur Model

Proses	Deskripsi
Penginputan Data	Data dimasukkan ke dalam model (teks, gambar, atau lainnya).
Pembagian Data dan Ekstraksi Fitur	Data dibagi dan fitur-fitur penting diekstrak (kata-kata atau frasa relevan).
Vektorisasi Fitur	Fitur diubah menjadi vektor melalui lapisan embedding (misalnya, Word2Vec).
Penerapan Convolutional Layers	Lapisan konvolusi digunakan untuk memahami pola dalam data.
Max Pooling	Lapisan max pooling diterapkan untuk meningkatkan efisiensi analisis.
Penggabungan Fitur	Hasil dari lapisan konvolusi dan max pooling digabungkan melalui lapisan concatenation.
Flattening	Data diratakan menjadi satu dimensi dengan lapisan flatten.
*Fully Connected Layer*s	Menggunakan lapisan fully connected untuk klasifikasi, serta fungsi aktivasi seperti Sigmoid untuk menentukan probabilitas kelas.

3.4. *Evaluasi*

Pada tahap evaluasi, model akan diuji menggunakan metrik seperti *confusion matrix*, *precision*, *recall*, F1-*score*, dan accuracy untuk memastikan keandalan serta konsistensi performa dari model identifikasi berita hoaks yang telah dikembangkan.

3.5. *Deployment*

Pada tahap deployment, model identifikasi berita hoaks akan diterapkan menggunakan Gradio untuk memudahkan integrasi dan interaksi dengan pengguna.

3.6. Teknik Pengumpulan Data

Data dikumpulkan melalui proses scraping pada platform media sosial Facebook dengan menggunakan kata kunci terkait politik, menggunakan perangkat lunak Apify.

3.7. Data dan Perangkat Penelitian

a. Data Primer
   Data primer dalam penelitian ini terdiri dari teks berita politik yang diklasifikasikan sebagai hoaks dan valid, dengan total 480 berita hoaks dan 520 berita valid.
b. Data Sekunder
   Data sekunder dalam penelitian ini mencakup jurnal-jurnal yang relevan dengan topik penelitian ini.

3.8. Objek Penelitian

Penelitian ini berfokus pada berita hoaks sebagai objek, dengan tujuan mengembangkan model identifikasi hoaks melalui pemodelan.

4. HASIL DAN PEMBAHASAN

Penelitian ini memfokuskan pada identifikasi berita hoaks menggunakan algoritma CNN dengan data dari Facebook. Proses pemodelan melibatkan pembagian dan pelatihan data, didukung oleh runtime T4 GPU di Google Colab untuk meningkatkan efisiensi pelatihan model.

4.1. Business Understanding

Tingginya penyebaran berita hoaks politik di Facebook menciptakan ketidakstabilan informasi. Penelitian ini bertujuan mengembangkan sistem identifikasi hoaks dengan model *Convolutional Neural Network* (CNN) untuk membedakan berita valid dari hoaks. Prosesnya meliputi pengumpulan dan pembersihan data, tokenisasi, vektorisasi teks, dan pelatihan model CNN untuk klasifikasi. Sistem ini diharapkan dapat meningkatkan akurasi deteksi berita hoaks dan mengurangi dampak negatif informasi palsu di media sosial.

4.2. Data Understanding

Pada tahap ini, kami memahami karakteristik dan kualitas data yang dikumpulkan untuk mengidentifikasi pola dan tren dalam teks berita. Ini juga membantu menentukan metode analisis yang tepat dan memastikan data memenuhi standar kualitas yang diperlukan untuk akurasi prediksi antara berita hoaks dan valid.

4.3. Pengumpulan Data

Data uji dan latih berupa teks politik dikumpulkan dari Facebook menggunakan teknik scraping. Data ini mencakup postingan dan artikel tentang isu politik seperti pilpres, kebijakan pemerintah, dan pileg dari halaman dan grup relevan. Proses scraping dilakukan sesuai etika dan ketentuan Facebook untuk memastikan data valid dan

representatif. Berikut adalah contoh data hasil scraping dari Facebook.

Tabel 4. *Sample Data*

No	Text	Label
1	Presiden menghormati putusan MK dan segala tuduhan pemohon kepada pemerintah tidak terbukti. Presiden Jokowi mengajak seluruh lapisan masyarakat bersatu membangun bangsa dan negara	Valid
2	Mangkrak di Era Ahok dan Anies Baswedan, Proyek Senilai Rp1,1 Triliun Justru Dibereskan Jokowi	Valid
3	MPR Gelar Sidang Istimewa Lengserkan Jokowi	Hoax

4.4. *WordCloud*

WordCloud digunakan untuk menganalisis dan memvisualisasikan frekuensi kata dalam teks berita, membantu mengidentifikasi topik utama yang mendominasi berita hoaks dan valid. Ini mengungkapkan pola kata khas dan berulang, mempermudah proses identifikasi. Berikut adalah visualisasi WordCloud untuk berita hoaks dan valid.

Gambar 1. *WordCloud* Berita Valid

Gambar 2. *WordCloud* Berita Hoaks

4.5. Data Preparation

Pada tahap ini, data berita dibersihkan untuk memastikan bahwa proses pemodelan sesuai dengan data yang sebenarnya.

4.6. *Labelling*

Setelah mengumpulkan data, sebanyak 1000 data, kemudian diberi label secara manual untuk memastikan akurasi dan relevansi antara berita hoaks dan valid.

4.7. Proses *Cleaning*

Proses ini mencakup langkah-langkah seperti mengubah teks menjadi huruf kecil, menghapus tanda baca, menghilangkan karakter non-alfanumerik, dan mengkonsolidasikan spasi berlebih untuk menghilangkan noise. Ini memastikan teks bebas dari gangguan yang dapat mempengaruhi pemodelan identifikasi berita hoaks dan membantu dalam mengidentifikasi pola bahasa yang membedakan berita hoaks dari berita valid.

Tabel 5. Data Sebelum Proses *Cleaning*

Index	Text
0	Presiden menghormati putusan MK dan segala tuduhan pemohon kepada pemerintah tidak terbukti. Presiden Jokowi mengajak seluruh lapisan masyarakat bersatu membangun bangsa dan negara
1	Mangkrak di Era Ahok dan Anies Baswedan, Proyek Senilai Rp1,1 Triliun Justru Dibereskan Jokowi
2	Sidang lanjutan sengketa hasil pilpres berlanjut pada Senin (1/4/2024). Timnas AMIN (Anies-Muhaimin) hadirkan 7 Ahli dan 11 Saksi. Salah satunya adalah Ahli Hukum dan Administrasi Ridwan.

Tabel 6. Data Setelah Proses *Cleaning*

Index	Text
0	Presiden menghormati putusan MK dan segala tuduhan pemohon kepada pemerintah tidak terbukti Presiden Jokowi mengajak seluruh lapisan masyarakat bersatu membangun bangsa dan negara
1	Mangkrak di Era Ahok dan Anies Baswedan Proyek Senilai Rp11 Triliun Justru Dibereskan Jokowi
2	Sidang lanjutan sengketa hasil pilpres berlanjut pada Senin 142024 Timnas AMIN AniesMuhaimin hadirkan 7 Ahli dan 11 Saksi Salah satunya adalah Ahli Hukum dan Administrasi Ridwan

4.8. *Case Folding*

Dalam penelitian ini, *Case Folding* digunakan untuk mengubah semua huruf teks berita menjadi huruf kecil. Ini memastikan perbedaan huruf besar-kecil tidak mempengaruhi analisis hoaks.

Tabel 7. Data Setelah Proses *Case Folding*

Index	Text
0	presiden menghormati putusan mk dan segala tuduhan pemohon kepada pemerintah tidak terbukti presiden jokowi mengajak seluruh lapisan masyarakat bersatu membangun bangsa dan negara
1	mangkrak di era ahok dan anies baswedan proyek senilai rp11 triliun justru dibereskan jokowi
2	sidang lanjutan sengketa hasil pilpres berlanjut pada senin 142024 timnas amin aniesmuhaimin hadirkan 7 ahli dan 11 saksi salah satunya adalah ahli hukum dan administrasi ridwan

4.9. *Tokenizing*

Tokenisasi diterapkan pada teks yang telah dibersihkan, mengubahnya menjadi urutan token yang mewakili kata-kata individual.

Tabel 8. Data Setelah Proses Tokenisasi

Index	Text
0	presiden,menghormati,putusan,mk,dan,segala, tuduhan,pemohon,kepada,pemerintah,tidak,ter bukti,presiden,jokowi,mengajak,seluruh,lapisa n,masyarakat,bersatu,membangun,bangsa,dan, negara
1	mangkrak,di,era,ahok,dan,anies,baswedan,pro yek, senilai,rp11,triliun,justru,dibereskan,jokowi
2	sidang,lanjutan,sengketa,hasil,pilpres,berlanjut ,pada,senin,142024,timnas,amin,aniesmuhaimi n,hadirkan,7,ahli,dan,11,saksi,salah,satunya,ad alah,ahli,hukum,dan,administrasi,ridwan

4.10. *StopWord Removal*

Penghilangan *StopWord* dilakukan untuk meningkatkan kualitas teks berita dengan menghapus kata-kata umum yang tidak penting, seperti "dan", "yang", dan "di".

Tabel 9. Data Setelah Proses *StopWord*

Index	text
0	presiden,menghormati,putusan,mk,tuduhan,pe mohon,pemerintah,terbukti,presiden,jokowi,m engajak,lapisan masyarakat,bersatu,membangun,bangsa,negara
1	mangkrak,era,ahok,anies,baswedan,proyek, senilai,rp11,triliun,dibereskan,jokowi
2	sidang,lanjutan,sengketa,hasil,pilpres,berlanjut ,senin,142024,timnas,amin,aniesmuhaimin,had irkan,7,ahli,11,saksi,salah,satunya,ahli,hukum, administrasi,ridwan

4.11. *Lemmatization*

```
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 66, 50)            123600

conv1d (Conv1D)              (None, 62, 256)           64256

max_pooling1d (MaxPooling1   (None, 31, 256)           0
D)

dropout (Dropout)            (None, 31, 256)           0

conv1d_1 (Conv1D)            (None, 27, 128)           163968

dropout_1 (Dropout)          (None, 27, 128)           0

conv1d_2 (Conv1D)            (None, 23, 64)            41024

max_pooling1d_1 (MaxPoolin   (None, 11, 64)            0
g1D)

dropout_2 (Dropout)          (None, 11, 64)            0

flatten (Flatten)            (None, 704)               0

dense (Dense)                (None, 256)               180480

dropout_3 (Dropout)          (None, 256)               0

dense_1 (Dense)              (None, 2)                 514

=================================================================
Total params: 573842 (2.19 MB)
Trainable params: 450242 (1.72 MB)
Non-trainable params: 123600 (482.81 KB)
_________________________________________________________________
```

Gambar 3. *Lemmatization*

Lemmatization dilakukan untuk mengubah kata-kata dalam teks berita ke bentuk dasarnya menggunakan *lemmatizer* Sastrawi. Ini mengurangi variasi kata dengan makna yang sama, meningkatkan akurasi pemodelan.

Tabel 10. Data Setelah Proses *Lemmatization*

Index	text
0	presiden,hormat,putus,mk,tuduh,mohon, perintah,bukti,presiden,jokowi,ajak,lapis,masy arakat,satu,bangun,bangsa,negara
1	presiden,hormat,putus,mk,tuduh,mohon,perint ah,bukti,presiden,jokowi,ajak,lapis,masyarakat ,satu,bangun,bangsa,negara
2	sidang,lanjut,sengketa,hasil,pilpres,lanjut,senin ,142024,timnas,amin,aniesmuhaimin,hadir,7,a hli,11,saksi,salah,satu,ahli,hukum,administrasi, ridwan

4.12. Word2Vec

Pembobotan kata dilakukan dengan model pre-trained Word2vec untuk mendapatkan representasi vektor kata dalam teks berita. Proses ini menghasilkan matriks embedding yang menangkap makna semantik dan hubungan antar kata.

```
[-1.4523931e-03  1.1152550e-02  3.6232341e-03 -7.6225805e-03
 -1.9099696e-03 -4.4140476e-03 -4.1085351e-03  1.4110599e-02
 -1.3763996e-02 -1.0189288e-02  6.4180110e-04 -1.6970551e-02
  2.2827871e-03  8.0928169e-03  1.1412946e-03 -1.6404290e-02
  1.1009407e-02 -1.7570650e-02  8.9561036e-03 -5.5280132e-03
 -2.4063846e-03 -4.2500966e-03 -4.7595619e-04 -4.0727821e-03
  4.2992947e-03 -6.5339878e-03 -3.6873298e-03  4.2988593e-03
 -4.0356820e-03 -7.1763201e-03  1.6317757e-02 -2.5574449e-03
  4.8596403e-03 -3.4798458e-04  4.3702121e-03  1.3598374e-02
  3.2266462e-03 -1.2973066e-02 -6.1477283e-03 -2.0146271e-02
 -1.7003558e-03 -1.8238679e-03 -1.0701154e-02 -9.1730505e-03
  2.5554460e-03 -1.0863844e-03 -8.9862188e-03 -4.2894413e-03
  1.7149930e-03  2.3373861e-03  4.7461139e-03 -4.6287961e-03
  8.1376908e-03  5.1172595e-03  5.4169828e-03  1.5099383e-03
  3.7194369e-03  8.5188905e-03  1.5380508e-03  2.0430679e-03
  3.6209931e-03  1.2596641e-03 -2.9136385e-03  5.1836483e-03
 -3.3713495e-03  5.9800451e-03  7.3749302e-03 -4.2421157e-03
 -8.1002219e-03  2.5037743e-04 -2.8303210e-03  1.1689748e-02
  9.6594766e-03  8.9852657e-04  1.4736274e-02  9.0647489e-05
 -9.6541662e-03  4.9047968e-03  3.1858177e-03  4.5022476e-03
  3.4259809e-03 -1.5957049e-03  3.0056085e-03  1.3168353e-02
 -7.8708939e-03  1.5703893e-03 -6.6316506e-04  9.0507101e-03
  3.4018541e-03  1.1350116e-02  1.8465489e-02  4.8491252e-03
 -2.8925003e-03  7.3326640e-03  1.6310526e-02  4.6494482e-03
  7.4757799e-03 -7.7596852e-03  1.2452570e-02  6.5492317e-03]
```

Gambar 4. *Word Embedding*

4.13. Modelling

Pada tahap ini, dilakukan perancangan model menggunakan arsitektur *Convolutional Neural Network* (CNN) untuk mengembangkan model klasifikasi berita hoaks.

4.14. Arsitektur Model

Model identifikasi berita hoaks dibangun menggunakan arsitektur *Convolutional Neural Network* (CNN) khusus untuk teks. Dimulai dengan lapisan Embedding yang mengonversi kata menjadi vektor berdimensi tetap menggunakan embedding yang sudah dilatih sebelumnya (trainable=False), diikuti oleh beberapa lapisan Convolutional dengan fungsi aktivasi ReLU untuk mengekstrak fitur penting dari teks. Setelah setiap lapisan convolutional, digunakan MaxPooling untuk mengurangi dimensi dan Dropout untuk mencegah overfitting dengan menonaktifkan neuron secara acak selama pelatihan.

Data kemudian di-flatten dan diteruskan ke lapisan Dense dengan ReLU serta regularisasi L2 untuk mencegah overfitting lebih lanjut, sebelum lapisan Dense terakhir dengan fungsi aktivasi Sigmoid untuk klasifikasi biner antara berita hoaks dan valid. Model dioptimalkan menggunakan Adam optimizer dengan learning rate yang disesuaikan dan dikompilasi dengan binary crossentropy sebagai fungsi loss, serta accuracy sebagai metrik evaluasi. Setelah arsitektur selesai dibangun, pelatihan model dilakukan dengan 100 epoch dan menggunakan early stopping untuk menghentikan pelatihan jika akurasi tidak meningkat, guna mencegah overfitting dan mempercepat proses pelatihan model. Berikut merupakan gambaran dari arsitektur model :

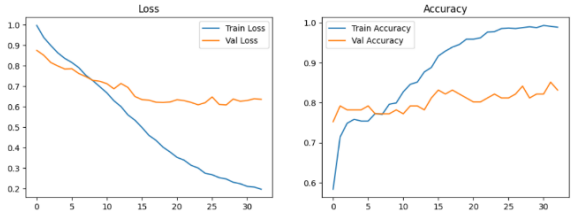
Gambar 5. Arsitektur Model

4.15. Uji Coba Skenario

Penelitian ini mencakup tiga skenario untuk menguji model dengan parameter terbaik dalam pengembangan sistem klasifikasi berita hoaks.

a. Skenario Pertama

Pada skenario pertama, model dengan arsitektur convolutional layer (256, 128, 64) dan *Fully Connected Layer* (dense 256) menggunakan optimizer Adam dan RMSprop, learning rate 0.001, kernel regularizer 0.001, dan 100 epoch. Dengan rasio dataset 90% pelatihan dan 10% pengujian, model mencatat akurasi 98,89% dan loss 0,1963 pada pelatihan. Namun, akurasi validasi turun menjadi 83,17% dengan loss 0,6360, mengindikasikan kemungkinan overfitting.



Gambar 6. Hasil Skenario Pertama

b. Skenario Kedua

Pada skenario ini, model dengan arsitektur convolutional layer (256, 128, 64) dan dropout 0,5, serta *Fully Connected Layer* (256) mencapai akurasi 92,53% pada pelatihan dan 81,09% pada pengujian, dengan loss 0,3325 untuk pelatihan dan 0,5523 untuk pengujian. Meskipun akurasi pelatihan tinggi, hasil ini menunjukkan model dalam kondisi goodfitting, dengan perbedaan akurasi yang wajar antara pelatihan dan pengujian.

Gambar 7. Hasil Skenario Kedua

c. Skenario Ketiga

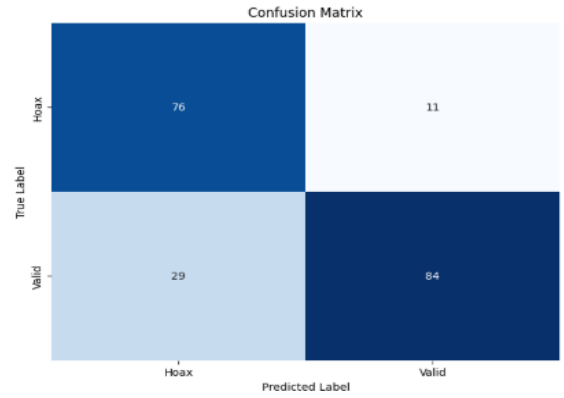
Pada skenario pertama, model dengan arsitektur convolutional layer (256, 128, 64) dan *Fully Connected Layer* (dense 256) serta dropout 0,5, menggunakan optimizer Adam dan RMSprop, learning rate 0,001, kernel regularizer 0,001, dan 100 epoch, menunjukkan hasil terbaik dengan rasio dataset 70% pelatihan dan 30% pengujian. Akurasi pelatihan mencapai 89,32%, sementara akurasi validasi adalah 76,82%. Training loss adalah 0,4462 dan validation loss 0,6825. Hasil ini menunjukkan potensi overfitting karena adanya perbedaan signifikan antara performa pelatihan dan validasi.

Rasio dataset	optimizer	Model skenario	Train		Test		Presisi	Recall	F1-Score
			Acc	Loss	Acc	Loss			
90% data latih, 10% data test	Adam	1	98	19	83	63	78	74	82
	RMSprop		64	45	60	76	76	78	67
80% data latih, 20% data test	Adam	2	92	33	81	55	69	76	80
	RMSprop		68	44	76	76	73	70	79
70% data latih, 30% data test	Adam	3	89	44	76	68	84	82	78
	RMSprop		65	54	71	67	64	76	67

Gambar 8. Hasil Skenario Ketiga

4.16. Evaluation

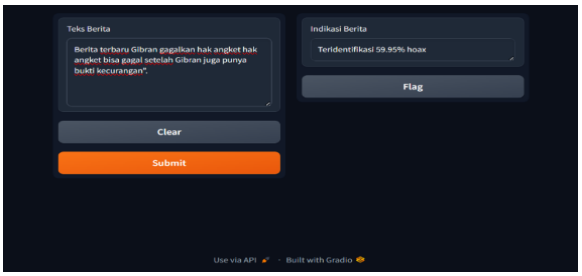
Evaluasi model **CNN** untuk identifikasi hoaks melibatkan metrik seperti akurasi, presisi, *recall*, F1-*score*, dan AUC-ROC untuk menilai kinerja klasifikasi. Pada skenario ini, model dengan arsitektur convolutional layer (256, 128, 64) dan dropout 0,5, serta Fully Connected Layer (256). Didapatkan hasil terbaik pada scenario 2 dengan akurasi mencapai 92,53% pada pelatihan dan 81,09% pada pengujian, dengan **loss** 0,33 untuk pelatihan dan 0,55 untuk pengujian. Meskipun akurasi pelatihan tinggi, hasil ini menunjukkan model dalam kondisi goodfitting, dengan perbedaan akurasi yang wajar antara pelatihan dan pengujian. Tiga skenario diuji, di mana skenario pertama dan ketiga menunjukkan overfitting, sedangkan skenario kedua menunjukkan performa yang lebih stabil, baik dari segi akurasi maupun loss. Evaluasi akhir dilakukan pada data pengujian menggunakan *confusion matrix* dan metrik seperti akurasi, presisi, *recall*, dan F1-*score* untuk memastikan efektivitas model dalam mengidentifikasi berita hoaks. Berikut merupakan hasil dari semua skenario dan hasil *confusion matrix*.



Gambar 9. *Confusion Matrix*

4.17. Deployment

Setelah memperoleh model terbaik, langkah selanjutnya adalah mengintegrasikannya ke dalam aplikasi web sederhana menggunakan Gradio. Gradio memungkinkan akses dan penggunaan model melalui browser web tanpa memerlukan pengetahuan teknis mendalam, sehingga pengguna dapat dengan mudah melihat hasil klasifikasi berita sebagai hoaks atau valid berdasarkan input yang diberikan. Berikut merupakan contoh hasil identifikasi.



Gambar 9. Hasil Identifikasi

5. KESIMPULAN DAN SARAN

Hasil dari penelitian ini menunjukkan bahwa model dengan arsitektur convolutional layer yang terdiri dari 4 filter (256, 128, 64), dropout 0,5 dan 0,3, serta Fully Connected Layer dengan 256 unit, memberikan hasil terbaik di antara tiga skenario yang diuji. Skenario kedua menunjukkan performa terbaik dengan akurasi 92,53% pada data pelatihan dan 81,09% pada data pengujian, serta nilai kerugian (loss) 0,33 dan 0,55. Dengan rasio pembagian data 80:20, model ini belajar dengan baik dari data pelatihan dan mampu menggeneralisasi dengan efektif pada data baru. Yang dapat membantu membedakan berita hoaks dan berita yang valid.

Untuk penelitian selanjutnya, disarankan agar menambah jumlah dataset dan ekplorasi berbagai arsitektur model untuk meningkatkan akurasi identifikasi berita hoaks dalam berbagai bahasa. Selain itu, memperluas konteks dari dataset politik ke kategori yang lebih umum akan membantu model dalam mengidentifikasi berita hoaks secara lebih luas.

DAFTAR PUSTAKA

[1] M. Rasidin, D. Witro, B. Z. Yanti, R. F. Purwaningsih, and W. Nurasih, "the Role of Government in Preventing the Spread of Hoax Related the 2019 Elections in Social Media," *Diakom J. Media dan Komun.*, vol. 3, no. 2, pp. 127–137, 2020, doi: 10.17933/diakom.v3i2.76.

[2] M. Ula, "Analisa Dan Deteksi Konten Hoax Pada Media Berita Indonesia Menggunakan Machine Learning," *J. Teknol. Terap. Sains 4.0*, vol. 1, no. 2, p. 229, 2020, doi: 10.29103/tts.v1i2.3263.

[3] S. Alyoubi, M. Kalkatawi, and F. Abukhodair, "The Detection of Fake News in Arabic Tweets Using Deep Learning," *Appl. Sci.*, vol. 13, no. 14, 2023, doi: 10.3390/app13148209.

[4] R. E. Hamzah and C. E. Putri, "Mengenal dan Mengantisipasi Hoax di Media Sosial pada Kalangan Pelajar," *J. Abdi MOESTOPO*, vol. 03, no. 01, pp. 9–12, 2020.

[5] M. R. S. Alfarizi, M. Z. Al-farish, M. Taufiqurrahman, G. Ardiansah, and M. Elgar, "Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning," *Karya Ilm. Mhs. Bertauhid (KARIMAH TAUHID)*, vol. 2, no. 1, pp. 1–6, 2023.

[6] R. A. Tilasefana and R. E. Putra, "Penerapan Metode Deep Learning Menggunakan Algoritma CNN Dengan Arsitektur VGG NET Untuk Pengenalan Cuaca," *J. Informatics Comput. Sci.*, vol. 05, no. 1, pp. 48–57, 2023.

[7] Y. Yunefri, Y. E. Fadrial, and S. Sutejo, "Chatbot Pada Smart Cooperative Oriented Problem Menggunakan Natural Language Processing dan Naive Bayes Classifier," *INTECOMS J. Inf. Technol. Comput. Sci.*, vol. 4, no. 2, pp. 131–140, 2021, doi: 10.31539/intecoms.v4i2.2704.

[8] V. R. Prasetyo, N. Benarkah, and V. J. Chrisintha, "Implementasi Natural Language Processing Dalam Pembuatan Chatbot Pada Program Information Technology Universitas Surabaya," *Teknika*, vol. 10, no. 2, pp. 114–121, 2021, doi: 10.34148/teknika.v10i2.370.

[9] A. Jinan, B. H. Hayadi, and U. P. Utama, "Klasifikasi Penyakit Tanaman Padi Mengunakan Metode Convolutional Neural Network Melalui Citra Daun (Multilayer Perceptron)," *J. Comput. Eng. Sci.*, vol. 1, no. 2, pp. 37–44, 2022.

[10] A. Khumaidi, "Data Mining for Predicting the Amount of Coffee Production Using Crisp-Dm Method," *J. Techno Nusa Mandiri*, vol. 17, no. 1, pp. 1–8, 2020, doi: 10.33480/techno.v17i1.1240.

[11] D. Normawati and S. A. Prayogi, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," *J. Sains Komput. Inform. (J-SAKTI*, vol. 5, no. 2, pp. 697–711, 2021.