

## LAMPIRAN

### Lampiran 1 Lembar Kesediaan Pembimbing

#### SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Ami Daelami Ichsana  
NIM : 20090133  
Program Studi : Teknik Informatika

Pihak Kedua

Nama : Dwi Intan Af'idah, S.T., Kom.  
Status : Dosen / Praktisi  
NIDN : 0620089303  
Jabatan Fungsional : Asisten Ahli  
Pangkat/Golongan : Penata Muda Tk.I/III

Pada hari ini Kamis tanggal 14 Maret 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing I/II Skripsi Pihak Pertama dengan **syarat adanya kemajuan pengerjaan skripsi yang dipresentasikan setiap dua minggu**. Adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

Tegal, 14 Maret 2024

Pihak Pertama



Ami Daelami Ichsana

Pihak Kedua



Dwi Intan Af'idah, S.T., Kom.

Mengetahui  
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliani, S.T., M.Kom.  
NIPY. 09.015.225

## SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Ami Daclami Ichsana  
NIM : 20090133  
Program Studi : Teknik Informatika

Pihak Kedua

Nama : Ardi Susanto, S.Kom., M.Cs.  
Status : Dosen / Praktisi  
NIDN : 0629109301  
Jabatan Fungsional : Asisten Ahli  
Pangkat/Golongan : Penata Muda Tk.I-III/b

Pada hari ini Jum'at tanggal 15 Maret 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing I/II Skripsi Pihak Pertama dengan **syarat Pihak Pertama wajib melakukan bimbingan Skripsi minimal 8 kali kepada Pihak Kedua**. Adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

Tegal, 15 Maret 2024

Pihak Pertama



Ami Daclami Ichsana

Pihak Kedua



Ardi Susanto, S.Kom., M.Cs.

Mengetahui  
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliani, S.T., M.Kom.  
NIPY. 09.015.225

## Lampiran 2. Surat Pernyataan Pengajuan HKI

### SURAT PERNYATAAN

Yang bertanda tangan di bawah ini, pemegang hak cipta:

1. Nama : Ami Daelami Ichsana  
Kewarganegaraan : Indonesia  
Alamat : Desa Purwahamba RT006/RW006, Kecamatan Suradadi,  
Kabupaten Tegal, 52182
2. Nama : Dwi Intan Af'idah, S.T., M.Kom.  
Kewarganegaraan : Indonesia  
Alamat : Desa Grinting RT003/RW001, Kecamatan Bulakamba,  
Kabupaten Brebes, 52253
3. Nama : Ardi Susanto, M.Cs.  
Kewarganegaraan : Indonesia  
Alamat : Margadana, Tegal

Dengan ini menyatakan bahwa:

1. Karya Cipta yang saya mohonkan:  
Berupa : Program Komputer  
Berjudul : Sistem Deteksi Judul Berita Clickbait dan Non-Clickbait di Media Online  
dengan Menggunakan Model IndoBert
  - Tidak meniru dan tidak sama secara esensial dengan Karya Cipta milik pihak lain atau obyektif kekayaan intelektual lainnya sebagaimana dimaksud dalam Pasal 68 ayat (2);
  - Bukan merupakan Ekspresi Budaya Tradisional sebagaimana dimaksud dalam Pasal 38;
  - Bukan merupakan Ciptaan yang tidak diketahui penciptanya sebagaimana dimaksud dalam Pasal 39;
  - Bukan merupakan hasil karya yang tidak dilindungi Hak Cipta sebagaimana dimaksud dalam Pasal 41 dan 42;
  - Bukan merupakan Ciptaan seni lukis yang berupa logo atau tanda pembeda yang digunakan sebagai merek dalam perdagangan barang/jasa atau digunakan sebagai lambang organisasi, badan usaha, atau badan hukum sebagaimana dimaksud dalam Pasal 65 dan;
  - Bukan merupakan Ciptaan yang melanggar norma agama, norma susila, ketertiban umum, pertahanan dan keamanan negara atau melanggar peraturan perundang-undangan sebagaimana dimaksud dalam Pasal 74 ayat (1) huruf d Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.
2. Sebagai pemohon mempunyai kewajiban untuk menyimpan asli contoh ciptaan yang dimohonkan dan harus memberikan apabila dibutuhkan untuk kepentingan penyelesaian sengketa perdata maupun pidana sesuai dengan ketentuan perundang-undangan.
3. Karya Cipta yang saya mohonkan pada Angka 1 tersebut di atas tidak pernah dan tidak sedang dalam sengketa pidana dan/atau perdata di Pengadilan.
4. Dalam hal ketentuan sebagaimana dimaksud dalam Angka 1 dan Angka 3 tersebut di atas saya / kami langgar, maka saya / kami bersedia secara sukarela bahwa:
  - a. Permohonan karya cipta yang saya ajukan dianggap ditarik kembali; atau

- b. Karya Cipta yang telah terdaftar dalam Daftar Umum Ciptaan Direktorat Hak Cipta, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum Dan Hak Asasi Manusia R.I dihapuskan sesuai dengan ketentuan perundang-undangan yang berlaku.
- c. Dalam hal kepemilikan Hak Cipta yang dimohonkan secara elektronik sedang dalam perkara dan/atau sedang dalam gugatan di Pengadilan maka status kepemilikan surat pencatatan elektronik tersebut ditangguhkan menunggu putusan Pengadilan yang berkekuatan hukum tetap.

Demikian Surat pernyataan ini saya/kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.



Pemegang Hak Cipta \*

(Dwi Intan Af'idah, S.T., M.Kom.)

Pemegang Hak Cipta \*  
10

(Ardi Susanto, M.Cs.)

Pemegang Hak Cipta \*

### Lampiran 3. Surat Pengalihan HKI

#### SURAT PENGALIHAN HAK CIPTA

Yang bertanda tangan di bawah ini :

1. Nama : Ami Daelami Ichsana  
Kewarganegaraan : Indonesia  
Alamat : Desa Purwahamba RT006/RW006, Kecamatan Suradadi  
Kabupaten Tegal, 52182
2. Nama : Dwi Intan Af'idah, S.T., M.Kom.  
Kewarganegaraan : Indonesia  
Alamat : Desa Grinting RT003/RW001, Kecamatan Bulakamba  
Kabupaten Brebes, 52253
- Nama : Ardi Susanto, M.Cs.  
Kewarganegaraan : Indonesia  
Alamat : Margadana, Tegal

Adalah **Pihak I** selaku pencipta, dengan ini menyerahkan karya ciptaan saya kepada :

Nama : Pusat Penelitian dan Pengabdian Masyarakat (P3M)  
Politeknik Harapan Bersama  
Alamat : Jl. Mataram No. 9 Pesurungan Lor Kota Tegal

Adalah **Pihak II** selaku Pemegang Hak Cipta berupa Program Komputer dengan judul "Sistem Deteksi Judul Berita Clickbait dan Non-Clickbait di Media Online dengan Menggunakan Model IndoBert". untuk didaftarkan di Direktorat Hak Cipta dan Desain Industri, Direktorat Jenderal Kekayaan Intelektual, Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia.

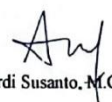
Demikianlah surat pengalihan hak ini kami buat, agar dapat dipergunakan sebagaimana mestinya.

Tegal, 16 Juli 2024

Pemegang Hak Cipta  
  
(Dr. Aldi Budi Riyanta, S.Si., M.T.)

Pencipta 10  
  
(Ami Daelami Ichsana)

  
(Dwi Intan Af'idah, S.T., M.Kom.)

  
(Ardi Susanto, M.Cs.)

**BeBerIn**

MANUAL BOOK

**DETEKSI JUDUL  
BERITA CLICKBAIT**

**2024**

Ami Daelami Ichsana  
Dwi Intan Af'idah, S.T., M.Kom.  
Ardi Susanto, S.Kom., M.Cs.

## **1. PENDAHULUAN**

### **1.1 Tujuan Pembuatan Dokumen**

Tujuan pembuatan dokumen ini adalah memberikan panduan yang jelas dan komprehensif untuk pengguna dalam menggunakan aplikasi BeBerIn, sehingga mereka dapat memaksimalkan manfaat aplikasi ini. Dokumen ini menjelaskan secara rinci fitur-fitur utama aplikasi, termasuk cara mengunggah artikel, menggunakan fungsi deteksi clickbait, dan mendapatkan analisis berita secara mudah dan cepat. Aplikasi BeBerIn dirancang untuk membantu pengguna dalam mengidentifikasi berita clickbait di media online, yang seringkali menyesatkan dan mengurangi kualitas informasi yang diterima publik. Dengan menggunakan teknologi deteksi clickbait, pengguna dapat dengan mudah memverifikasi apakah judul berita mengandung unsur clickbait atau tidak.

### **1.2 Deskripsi Umum Sistem**

#### **1.2.1 Deskripsi Umum Aplikasi**

BeBerIn adalah sebuah aplikasi inovatif yang dirancang untuk mendeteksi dan menganalisis berita clickbait di media online. Dengan meningkatnya jumlah berita yang menyesatkan dan memanipulasi pembaca, aplikasi ini hadir sebagai solusi untuk membantu pengguna mengidentifikasi dan menghindari berita-berita semacam itu. BeBerIn menggunakan teknologi canggih untuk menganalisis judul berita, memberikan penilaian apakah suatu berita mengandung unsur clickbait atau tidak.

#### **1.2.2 Deskripsi Umum Kebutuhan Aplikasi**

BeBerIn adalah aplikasi web untuk mendeteksi dan menganalisis berita clickbait di media online menggunakan teknologi artificial intelligence. Aplikasi ini menggunakan Flask untuk backend, HTML, CSS, dan JavaScript untuk frontend, serta SQLite untuk penyimpanan data. Aplikasi ini memungkinkan pengguna untuk menginputkan judul berita atau mengunggah file excel yang berisi judul berita yang akan dilabeli secara otomatis. BeBerIn dirancang untuk skalabilitas dan performa tinggi, memastikan analisis cepat dan akurat serta pengalaman pengguna yang optimal dalam mengidentifikasi berita clickbait. Dengan BeBerIn, pengguna dapat meningkatkan literasi media dan menghindari berita yang menyesatkan.

### **1.3 Deskripsi Dokumen**

#### **1. BAB I**

Memperkenalkan konsep dan tujuan aplikasi BeBerIn, serta menjelaskan teknologi artificial intelligence dengan menggunakan metode transformers yaitu IndoBert yang digunakan untuk mendeteksi dan menganalisis berita clickbait di media online.

#### **2. BAB II**

Menjelaskan secara rinci user interface aplikasi, termasuk fitur-fitur utama seperti input judul berita, mengunggah file excel yang berisi judul berita, dan tampilan hasil deteksi.

#### **3. BAB III**

Fokus pada teknologi yang digunakan dalam pengembangan aplikasi, meliputi Flask untuk backend, HTML, CSS, dan JavaScript untuk frontend, serta integrasi SQLite untuk pengelolaan data.

### **2. PERANGKAT YANG DIBUTUHKAN**

#### **2.1 Perangkat Lunak**

1. Windows sebagai Operating System.
2. Google Chrome sebagai Tools Penjelajah (Web Browser)
3. Peralatan Jaringan Internet

#### **2.2 Perangkat Keras**

1. Flask (BackEnd)
2. HTML, CSS, dan Java Script (FrontEnd)
3. SQLite (Database)
4. Python (Language Programming Model)
5. Library Python

### **3. MENU DAN CARA PENGGUNAAN**

#### **3.1 Struktur Menu**

Adapun struktur menu pada website Sistem Deteksi Judul Berita Clickbait di Media Online sebagai berikut :

- a. Home
- b. Deteksi
- c. Report



d. Tentang Kami

Adapun struktur menu pada website dashboard Sistem Deteksi Judul Berita Clickbait dengan menggunakan streamlit sebagai berikut:

- a. GridID
- b. Detikcom
- c. Report

### 3.2 Penggunaan Aplikasi

Pada bagian ini akan dijelaskan mengenai menu-menu yang ada pada website Sistem Deteksi Judul Berita Clickbait pada masing-masing user role.

#### 1. Pengguna Website BeBerIn

Buka website “BeBerIn” dengan menggunakan web browser dengan url yang telah disediakan.

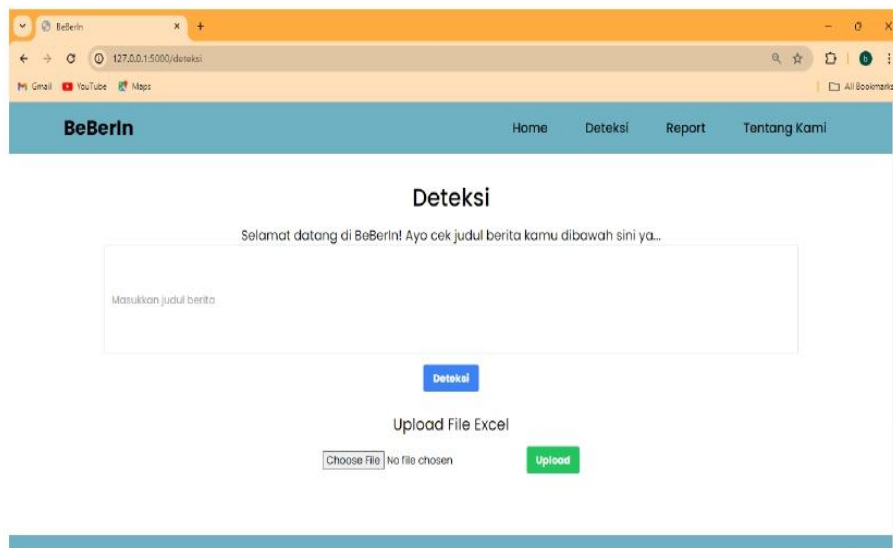
##### a. Home

Halaman utama di BeBerIn. Pada halaman ini, pengguna dapat melihat informasi tentang pencegahan penyebaran clickbait dan mendukung informasi akurat. Ada beberapa menu di bagian atas seperti "Home", "Deteksi", "Report", dan "Tentang Kami". Pada bagian bawah, terdapat informasi tentang apa itu berita clickbait serta beberapa contoh berita clickbait. Informasi ini membantu pengguna memahami karakteristik clickbait dan bagaimana mengidentifikasinya



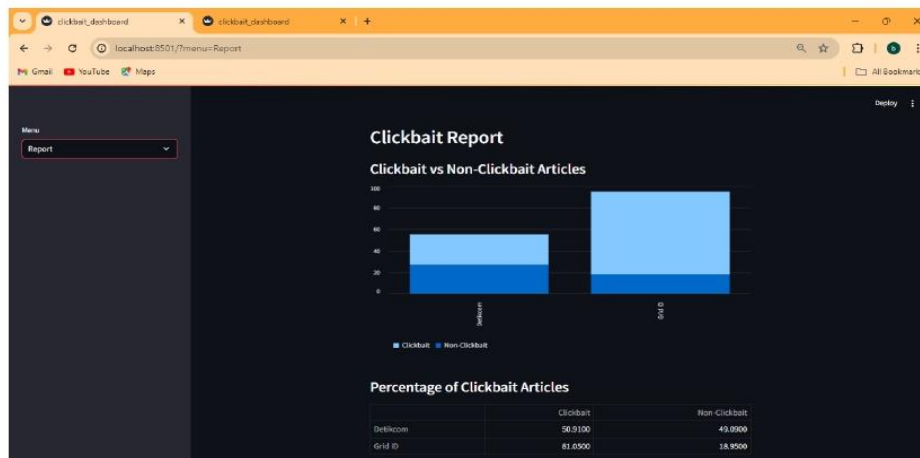
## b. Deteksi

Pada halaman ini, pengguna dapat memasukkan judul berita untuk mendeteksi apakah berita tersebut merupakan clickbait atau bukan. Terdapat kolom untuk memasukkan judul berita dan tombol "Deteksi" untuk memulai proses deteksi. Selain itu, pengguna juga dapat mengunggah file Excel dengan menggunakan tombol "Upload File Excel" untuk memeriksa beberapa judul berita sekaligus.



## c. Report

Pada bagian "Report", terdapat visualisasi data yang menunjukkan perbandingan antara artikel clickbait dan non-clickbait pada dua website yaitu GridID dan Detikcom. Grafik batang pada tampilan ini memberikan informasi mengenai jumlah artikel clickbait dibandingkan dengan artikel non-clickbait. Selain itu, terdapat grafik persentase yang menunjukkan proporsi artikel clickbait terhadap keseluruhan artikel.



#### d. Tentang Kami

Di bagian "Tentang Kami", pengguna dapat menemukan informasi mengenai tujuan dan visi dari aplikasi ini. Halaman ini menjelaskan latar belakang dan alasan pembuatan aplikasi, serta tim pengembang yang terlibat. Pengguna dapat mempelajari lebih lanjut mengenai misi kami untuk menyediakan alat yang dapat mendeteksi dan menganalisis artikel clickbait, serta bagaimana aplikasi ini dapat membantu meningkatkan kualitas informasi yang tersedia bagi publik.

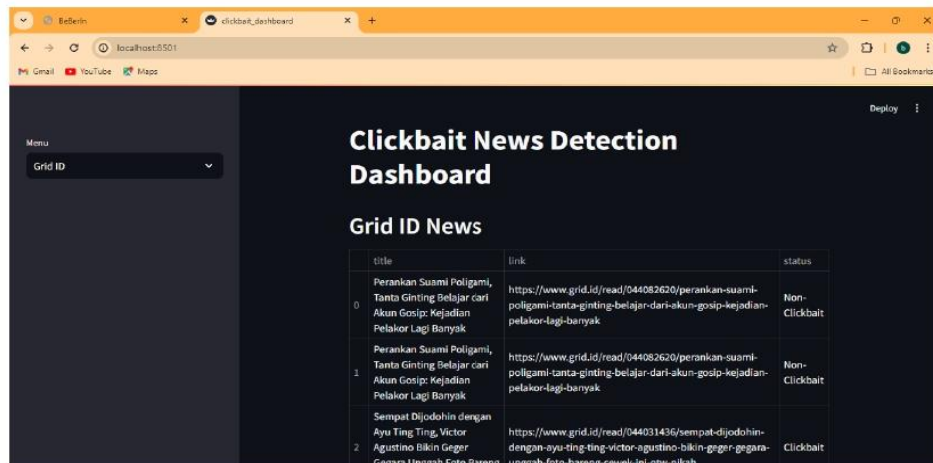


## 2. Penggunaan Website Dashboard BeBerIn

Buka website "BeBerIn" dengan menggunakan web browser dengan url yang telah disediakan.

### a. GridID

Pada halaman ini merupakan halaman menu GridID News. Menu ini berfungsi untuk mendeteksi dan menampilkan status clickbait dari berbagai artikel berita yang diterbitkan oleh Grid ID. Pada menu ini, terdapat tabel yang terdiri dari beberapa kolom, yaitu "judul," "link," "tanggal," dan "status". Kolom "judul" berisi judul-judul artikel berita, sedangkan kolom "link" memuat URL atau tautan yang mengarah langsung ke artikel tersebut. Kolom "tanggal" menunjukkan tanggal publikasi masing-masing artikel berita, memberikan informasi tambahan mengenai waktu penerbitan artikel tersebut. Terakhir, kolom "status" menunjukkan apakah artikel tersebut terdeteksi sebagai clickbait atau bukan clickbait berdasarkan model deteksi yang digunakan. Dashboard ini memberikan tampilan yang jelas dan terstruktur, memudahkan pengguna untuk melihat dan menganalisis status clickbait dari artikel-artikel berita yang ada di Grid ID News.

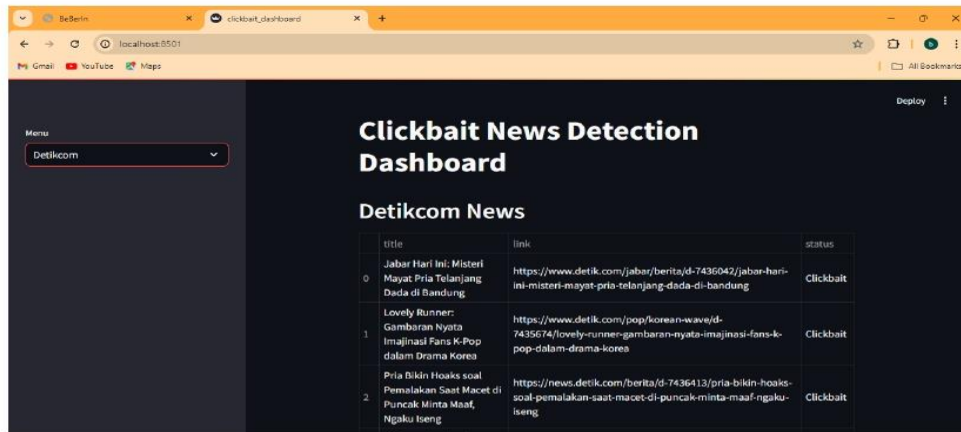


	title	link	status
0	Perankan Suami Poligami, Tanta Ginting Belajar cari Akun Gosip: Kejadian Pelakor Lagi Banyak	<a href="https://www.grid.id/read/044082620/perankan-suami-poligami-tanta-ginting-belajar-dari-akun-gosip-kejadian-pelakor-lagi-banyak">https://www.grid.id/read/044082620/perankan-suami-poligami-tanta-ginting-belajar-dari-akun-gosip-kejadian-pelakor-lagi-banyak</a>	Non-Clickbait
1	Perankan Suami Poligami, Tanta Ginting Belajar cari Akun Gosip: Kejadian Pelakor Lagi Banyak	<a href="https://www.grid.id/read/044082620/perankan-suami-poligami-tanta-ginting-belajar-dari-akun-gosip-kejadian-pelakor-lagi-banyak">https://www.grid.id/read/044082620/perankan-suami-poligami-tanta-ginting-belajar-dari-akun-gosip-kejadian-pelakor-lagi-banyak</a>	Non-Clickbait
2	Sempat Dijodohin dengan Ayu Ting Ting, Victor Agustino Bikin Geger-gegara Unyah Foto Bareng	<a href="https://www.grid.id/read/044031436/sempat-dijodohin-dengan-ayu-ting-ting-victor-agustino-bikin-geger-gegara-unyah-foto-bareng-cewek-ini-otw-nikah">https://www.grid.id/read/044031436/sempat-dijodohin-dengan-ayu-ting-ting-victor-agustino-bikin-geger-gegara-unyah-foto-bareng-cewek-ini-otw-nikah</a>	Clickbait

### b. DetikCom

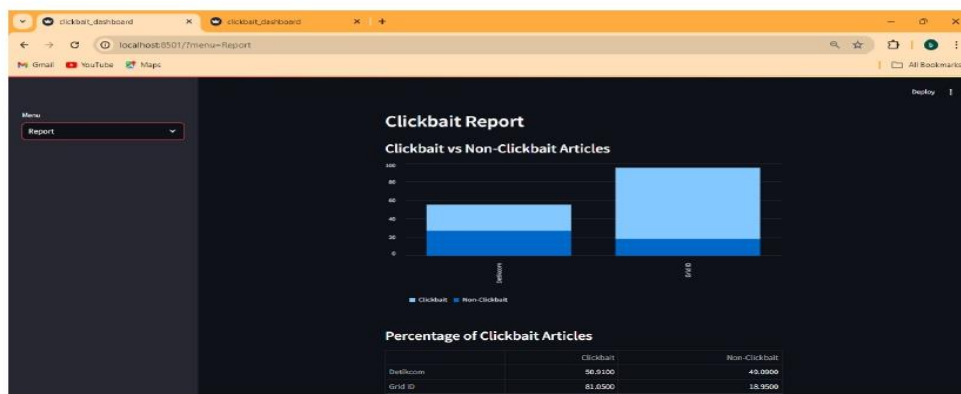
Pada halaman ini merupakan halaman menu Detikcom News. Menu ini berfungsi untuk mendeteksi dan menampilkan status clickbait dari berbagai artikel berita yang diterbitkan oleh Detikcom. Pada menu ini, terdapat tabel yang terdiri dari beberapa kolom, yaitu "judul," "link," "tanggal," dan "status". Kolom "judul" berisi judul-judul artikel berita, sedangkan kolom "link" memuat URL atau tautan yang mengarah langsung ke artikel tersebut. Kolom "tanggal" menunjukkan tanggal publikasi masing-masing artikel berita, memberikan informasi tambahan

mengenai waktu penerbitan artikel tersebut. Terakhir, kolom "status" menunjukkan apakah artikel tersebut terdeteksi sebagai clickbait atau bukan clickbait berdasarkan model deteksi yang digunakan. Dashboard ini memberikan tampilan yang jelas dan terstruktur, memudahkan pengguna untuk melihat dan menganalisis status clickbait dari artikel-artikel berita yang ada di Detikcom.



### c. Report

Pada bagian "Report", terdapat visualisasi data yang menunjukkan perbandingan antara artikel clickbait dan non-clickbait pada dua website yaitu GridID dan Detikcom. Grafik batang pada tampilan ini memberikan informasi mengenai jumlah artikel clickbait dibandingkan dengan artikel non-clickbait. Selain itu, terdapat grafik persentase yang menunjukkan proporsi artikel clickbait terhadap keseluruhan artikel.



**BeBerIn**

DOKUMEN TEKNIKAL

**DETEKSI JUDUL  
BERITA CLICKBAIT**

**2024**

Ami Daelami Ichsana  
Dwi Intan Af'idah, S.T., M.Kom.  
Ardi Susanto, S.Kom., M.Cs.

## **Profil**

BeBerIn adalah sebuah website yang dirancang untuk mendeteksi dan menganalisis berita clickbait di media online. Aplikasi ini menggunakan teknologi Artificial Intelligent dengan menggunakan algoritma indobert untuk mengidentifikasi pola clickbait dalam artikel berita. BeBerIn menyediakan berbagai fitur penting yang mendukung proses unggah file excel berisi judul berita yang akan dilabeli secara otomatis, input judul berita secara langsung, dan tampilan hasil deteksi. Website ini memberikan solusi terstruktur untuk membantu pengguna menghindari berita yang menyesatkan.

## **Latar Belakang**

Berita clickbait telah menjadi fenomena yang meresahkan dalam dunia media online, seringkali mengorbankan kualitas informasi demi menarik perhatian pembaca. Dampak dari berita clickbait tidak hanya menyesatkan pembaca, tetapi juga merusak reputasi media yang terpercaya. Dalam era informasi yang semakin cepat dan mudah diakses, penting bagi pengguna untuk memiliki alat yang dapat membantu mereka menyaring dan mengidentifikasi berita yang akurat dan dapat dipercaya. Banyak pengguna yang kesulitan membedakan antara berita yang valid dan yang mengandung clickbait, terutama karena taktik yang digunakan dalam judul berita sangat bervariasi dan semakin canggih. Oleh karena itu, pengembangan aplikasi BeBerIn menjadi sangat relevan untuk memberikan solusi terhadap tantangan ini.

BeBerIn dirancang untuk membantu pengguna mengenali elemen-elemen clickbait dalam berita secara cepat dan akurat. Dengan teknologi AI, aplikasi ini mampu menganalisis pola-pola dalam judul berita untuk menentukan apakah berita tersebut mengandung clickbait. Selain itu, aplikasi ini juga menawarkan fitur pelaporan yang memberikan informasi mendalam mengenai hasil analisis, membantu pengguna dalam memahami karakteristik berita yang mereka baca. Aplikasi BeBerIn juga dirancang dengan mempertimbangkan keamanan dan privasi pengguna, meskipun tidak memiliki fitur login, data yang diunggah tetap diperlakukan dengan standar keamanan tinggi. Selain itu, aplikasi ini dioptimalkan untuk skalabilitas dan performa, memastikan bahwa analisis dapat dilakukan dengan cepat dan efisien bahkan dengan jumlah pengguna yang besar. Dengan memanfaatkan teknologi AI terbaru dan mengintegrasikan berbagai fitur yang relevan, BeBerIn diharapkan mampu menjadi solusi yang handal dalam meningkatkan literasi media dan membantu pengguna menghindari berita clickbait di era digital ini.

## **Manfaat**

Adapun manfaat yang diperoleh dari penelitian ini ada tiga, diantaranya sebagai berikut:

1. Meningkatkan efisiensi dan efektivitas dalam mengidentifikasi dan menganalisis berita clickbait di media online.
2. Memperkuat literasi media pengguna dengan menyediakan alat yang membantu menghindari berita yang menyesatkan.
3. Mengasah kreativitas penulis dalam menghadapi tantangan teknis, seperti mengintegrasikan fitur kompleks ke dalam sistem dan merancang antarmuka pengguna yang intuitif.

## **Spesifikasi Teknis**

Spesifikasi teknis meliputi:

### 1. Perangkat Keras

Perangkat keras yang digunakan untuk pengembangan aplikasi BeBerIn ini adalah sebagai berikut:

- a. Laptop HP Processor 13th Gen Intel(R) Core(TM) i3-1315U 1.20 GHz
- b. SSD 512GB
- c. RAM 8GB

### 2. Perangkat Lunak

Perangkat lunak yang digunakan untuk pengembangan aplikasi BeBerIn ini adalah sebagai berikut:

- a. Visual Studio Code
- b. Google Colab
- c. Flask Framework
- d. Google Chrome
- e. SQLite
- f. HTML, CSS, dan JavaScript

### 3. Source Code

Beberapa *source code* dalam membangun website BeBerIn dijelaskan sebagai berikut:

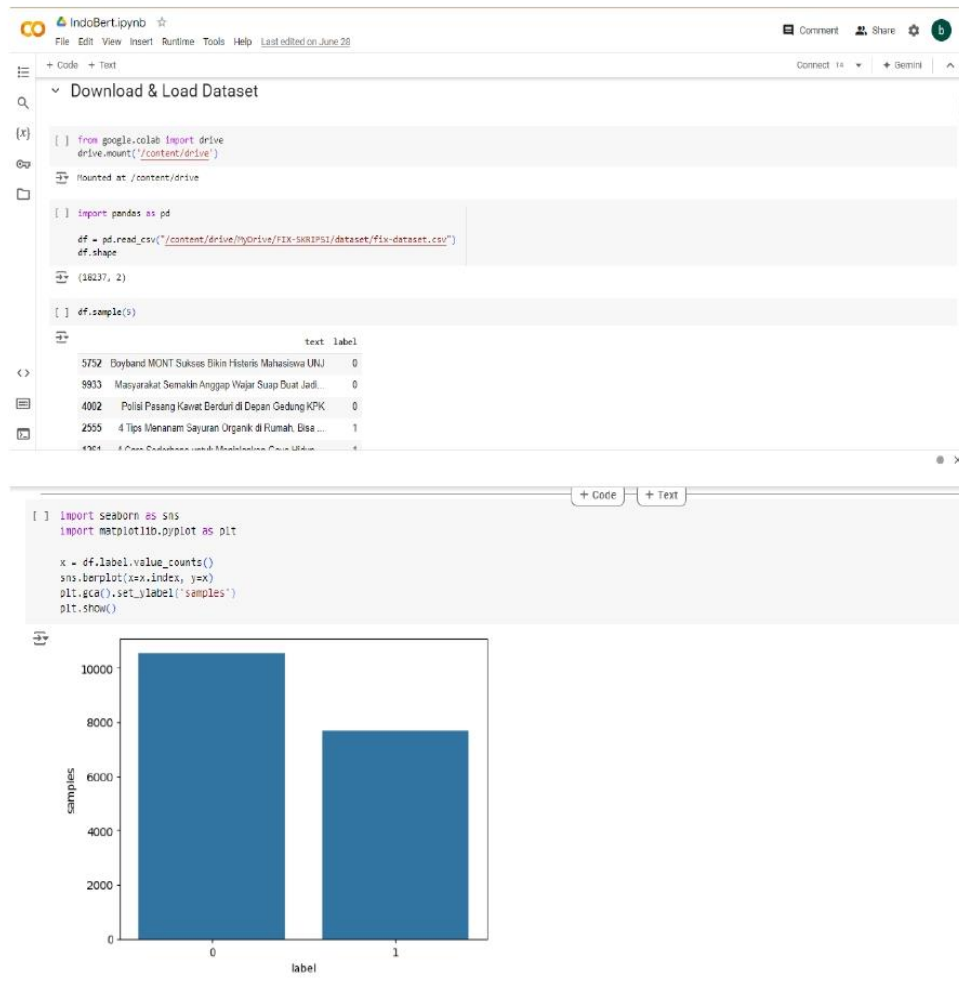


a. Model AI

Model AI yang digunakan dalam aplikasi BeBerIn adalah IndoBERT, yang berfungsi sebagai inti dari deteksi dan analisis berita clickbait. Model ini memiliki peran yaitu mengelola logika deteksi clickbait dengan memproses input data dari pengguna, berperan sebagai perantara antara data (model) dan antarmuka pengguna (view), memastikan bahwa analisis yang dihasilkan akurat dan relevan, dan menangani proses judul berita, serta mengirimkan hasil deteksi clickbait kembali kepada pengguna dengan cepat dan efisien.

1) Data load dan Visualisasi

Pada bagian load data dan visualisasi untuk memuat dataset dari Google Drive dan menampilkannya menggunakan library pandas serta terdapat visualisasi jumlah data antara label 0 yaitu non-clickbait dan 1 yaitu clickbait.



## 2) Preprocessing

Preprocessing adalah langkah-langkah yang dilakukan untuk mempersiapkan data mentah agar dapat digunakan oleh model machine learning atau algoritma lainnya. Dalam konteks Natural Language Processing (NLP), preprocessing biasanya melibatkan berbagai teknik untuk membersihkan dan mengonversi teks sehingga dapat dimasukkan ke dalam model. Berikut adalah penjelasan langkah-langkah preprocessing yang dilakukan dalam kode tersebut:

- Tokenisasi

Bagian ini menampilkan kalimat asli, hasil tokenisasi, dan konversi token ke ID-nya. Tokenisasi adalah proses memecah kalimat menjadi bagian-bagian yang lebih kecil yang disebut token.

```
[.] print("Original: ", sentences[0])
print("Tokenized: ", tokenizer.tokenize(sentences[0]))
print("Token IDs: ", tokenizer.convert_tokens_to_ids(tokenizer.tokenize(sentences[0])))
```

Original: Purnomo Serahkan Surat Mundur dari Pilkada ke DPC PDIP Solo  
Tokenized: ['pur', '##nom', '##o', 'sera', '##nkan', 'surat', 'mundu', '##', 'dari', 'pil', '##kada', 'ke', 'dp', '##c', 'pd', '##io', 'solo']  
Token IDs: [27327, 12883, 10132, 13796, 24416, 31431, 26124, 10131, 10443, 89449, 66001, 11009, 57474, 10261, 47843, 25113, 11174]

- Encoding Kalimat

Setiap kalimat di-encode menjadi daftar ID token, termasuk menambahkan token spesial jika diperlukan. Ini mengonversi teks ke dalam bentuk numerik yang dapat digunakan oleh model.

```
input_ids = []
for sent in sentences:
    encoded_sent = tokenizer.encode(
        sent,
        add_special_tokens = True
    )
    input_ids.append(encoded_sent)
print("Original: ", sentences[0])
print("Token IDs: ", input_ids[0])
```

Original: Purnomo Serahkan Surat Mundur dari Pilkada ke DPC PDIP Solo  
Token IDs: [101, 27327, 12883, 10132, 13796, 24416, 31431, 26124, 10131, 10443, 89449, 66001, 11009, 57474, 10261, 47843, 25113, 11174, 102]

- Padding dan Truncating

Semua kalimat dipotong atau ditambahkan padding agar memiliki panjang yang sama, yaitu 64 token. Padding berguna agar semua input memiliki panjang yang sama ketika dimasukkan ke dalam model.

```
[ ] from keras.preprocessing.sequence import pad_sequences

MAX_LEN = 64

print("Padding/truncating all sentences to %d values" % MAX_LEN)
print("Padding token: "[:]", ID: {:}").format(tokenizer.pad_token, tokenizer.pad_token_id))

input_ids = pad_sequences(input_ids, maxlen=MAX_LEN, dtype='long', value=0, truncating='post', padding='post')

print("Done")
```

↳ Padding/truncating all sentences to 64 values  
 Padding token: "[PAD]", ID: 0  
 Done

### 3) Persiapan Data

Proses ini adalah persiapan data sebelum dimasukkan ke dalam model Transformer, yang mencakup pembagian data, konversi data ke tensor, dan pembuatan DataLoader untuk pelatihan, validasi, dan pengujian model. Berikut adalah beberapa detail tambahan untuk menghubungkan kode tersebut dengan model Transformer:

- **Pembagian Data Latih dan Pengujian**

Data dibagi menjadi set pelatihan dan set pengujian dengan perbandingan 90% untuk pelatihan dan 10% untuk pengujian. `random_state` digunakan untuk memastikan hasil yang konsisten setiap kali kode dijalankan. Set pelatihan dibagi lagi menjadi set pelatihan dan set validasi dengan perbandingan 85% untuk pelatihan dan 15% untuk validasi. `random_state` digunakan untuk memastikan hasil yang konsisten.

```
[ ] from sklearn.model_selection import train_test_split

train_input, test_input, train_labels, test_labels = train_test_split(input_ids,
                                                                    labels,
                                                                    random_state=2017,
                                                                    test_size=0.1)

train_mask, test_mask, _, _ = train_test_split(attention_mask,
                                              labels,
                                              random_state=2017,
                                              test_size=0.1)

train_input, validation_input, train_labels, validation_labels = train_test_split(train_input,
                                                                                  train_labels,
                                                                                  random_state=2018,
                                                                                  test_size=0.15)

train_mask, validation_mask, _, _ = train_test_split(train_mask,
                                                    train_mask,
                                                    random_state=2018,
                                                    test_size=0.15)
```

- **Pembagian Data Pelatihan Menjadi Set Pelatihan dan Validasi**

Bagian ini menampilkan bentuk (`shape`) dari setiap set data (pelatihan, validasi, dan pengujian) untuk memastikan bahwa data telah dibagi dengan benar.

```

▶ import numpy as np
print("== Train ==")
print("Input: ", train_input.shape)
print("Label: ", train_labels.shape)
print("Mask: ", np.array(train_mask).shape)

print("\n== Validation ==")
print("Input: ", validation_input.shape)
print("Label: ", validation_labels.shape)
print("Mask: ", np.array(validation_mask).shape)

print("\n== Test ==")
print("Input: ", test_input.shape)
print("Label: ", test_labels.shape)
print("Mask: ", np.array(test_mask).shape)

```

- Konversi Data Menjadi Tensor PyTorch

Data input, label, dan mask dikonversi menjadi tensor PyTorch agar dapat digunakan dalam DataLoader.

```

[ ] train_input = torch.tensor(train_input)
    train_labels = torch.tensor(train_labels)
    train_mask = torch.tensor(train_mask)

    validation_input = torch.tensor(validation_input)
    validation_labels = torch.tensor(validation_labels)
    validation_mask = torch.tensor(validation_mask)

    test_input = torch.tensor(test_input)
    test_labels = torch.tensor(test_labels)
    test_mask = torch.tensor(test_mask)

```

- Pembuatan DataLoader untuk Set Pelatihan, Validasi, dan Pengujian

DataLoader digunakan untuk memuat data dalam batch, yang berguna untuk pelatihan model dalam iterasi. RandomSampler digunakan untuk set pelatihan agar data diload secara acak setiap epoch, sedangkan SequentialSampler digunakan untuk set validasi dan pengujian agar data diload secara urut.

```

from torch.utils.data import TensorDataset, DataLoader, RandomSampler, SequentialSampler

batch_size = 32

train_data = TensorDataset(train_input, train_mask, train_labels)
train_sampler = RandomSampler(train_data)
train_dataloader = DataLoader(train_data, sampler=train_sampler, batch_size=batch_size)

validation_data = TensorDataset(validation_input, validation_mask, validation_labels)
validation_sampler = SequentialSampler(validation_data)
validation_dataloader = DataLoader(validation_data, sampler=validation_sampler, batch_size=batch_size)

test_data = TensorDataset(test_input, test_mask, test_labels)
test_sampler = SequentialSampler(test_data)
test_dataloader = DataLoader(test_data, sampler=test_sampler, batch_size=batch_size)

```

#### 4) Instansiasi dan Pre-trained Model

- Konfigurasi Model Bert

Pada bagian ini, model BERT untuk klasifikasi sekumpulan sekuens diinstansiasi dari model pre-trained bert-base-multilingual-uncased. Model dikonfigurasi untuk klasifikasi biner (dua label) dan diatur untuk tidak mengeluarkan attention atau hidden states. Model kemudian dipindahkan ke GPU dengan model.cuda().

```

[ ] from transformers import BertForSequenceClassification, AdamW, BertConfig

model = BertForSequenceClassification.from_pretrained(
    "bert-base-multilingual-uncased",
    num_labels = 2,
    output_attentions = False,
    output_hidden_states = False
)

model.cuda()

```

- Menampilkan Parameter Model

Parameter model ditampilkan untuk memberikan wawasan tentang struktur model, termasuk layer embedding, layer Transformer pertama, dan layer output.

```

[ ] params = list(model.named_parameters())

print("The BERT model has {:} different named parameters.".format(len(params)))

print("==== Embedding Layer ====")
for p in params[0:5]:
    print("{:<60} {:>12}".format(p[0], str(tuple(p[1].size()))))

print("==== First Transformers ====")
for p in params[5:21]:
    print("{:<60} {:>12}".format(p[0], str(tuple(p[1].size()))))

print("==== Output Layer ====")
for p in params[-4:]:
    print("{:<60} {:>12}".format(p[0], str(tuple(p[1].size()))))

```

- Konfigurasi Optimizer

Optimizer AdamW diatur untuk melatih model dengan learning rate dan epsilon yang ditentukan. AdamW adalah versi yang disarankan dari Adam optimizer untuk model Transformer.

```
optimizer = AdamW(  
    model.parameters(),  
    lr = 2e-5,  
    eps = 1e-8  
)
```

- Mengatur Scheduler

Scheduler untuk mengatur learning rate selama pelatihan diatur menggunakan `get_linear_schedule_with_warmup`, yang secara linier mengurangi learning rate setelah beberapa langkah awal (warmup steps).

```
[ ] from transformers import get_linear_schedule_with_warmup  
  
epochs = 5  
  
total_steps = len(train_dataloader) * epochs  
  
scheduler = get_linear_schedule_with_warmup(optimizer,  
                                             num_warmup_steps = 0,  
                                             num_training_steps = total_steps)
```

- Definisi Fungsi untuk Menghitung Akurasi

Fungsi `flat_accuracy` untuk menghitung akurasi dari prediksi model dibandingkan dengan label sebenarnya.

```
import numpy as np  
  
def flat_accuracy(preds, labels):  
    pred_flat = np.argmax(preds, axis=1).flatten()  
    labels_flat = labels.flatten()  
    return np.sum(pred_flat == labels_flat) / len(labels_flat)
```

## 5) Training Model

- Penetapan Seed untuk Reprodusibilitas

Menetapkan seed untuk berbagai library (random, numpy, torch) untuk memastikan hasil pelatihan yang dapat direproduksi.

```

import random

seed_val = 42

random.seed(seed_val)
np.random.seed(seed_val)
torch.manual_seed(seed_val)
torch.cuda.manual_seed_all(seed_val)

loss_values = []

for epoch_i in range(0, epochs):

    # -----
    #           Training
    # -----

    print("----- Epoch {} / {} -----".format(epoch_i+1, epochs))
    print("Training...")

    t0 = time.time()

    total_loss = 0

```

- Loop Epoch untuk Pelatihan dan Validasi

Pada bagian pelatihan untuk melatih model pada data pelatihan, menghitung dan mencatat loss, mengupdate parameter model, dan menampilkan progress. Sedangkan validasi untuk mengevaluasi model pada data validasi, menghitung akurasi, dan menampilkan hasilnya.

```

for epoch_i in range(e, epochs):

    # =====
    #           Training
    # =====

    print("===== Epoch {} / {} =====".format(epoch_i+1, epochs))
    print("Training...")

    t0 = time.time()

    total_loss = 0

    model.train()

    # For each batch of training data
    for step, batch in enumerate(train_dataloader):

        # Progress update every 40 batches
        if step % 40 == 0 and not step == 0:
            elapsed = format_time(time.time() - t0)

            print("Batch {:>5}, of {:>5},. Elapsed: {}".format(step, len(train_dataloader), elapsed))

        b_input_ids = batch[0].to(device)
        b_input_mask = batch[1].to(device)
        b_labels = batch[2].to(device)

        model.zero_grad()

        outputs = model(b_input_ids,
                        token_type_ids=None,

```

- Visualisasi Training Loss

Pada bagian ini visualisasi training loss menggunakan Matplotlib dan Seaborn untuk memvisualisasikan training loss selama beberapa epoch.

```
import matplotlib.pyplot as plt
%matplotlib inline

import seaborn as sns

sns.set(style='darkgrid')

sns.set(font_scale=1.5)
plt.rcParams["figure.figsize"] = (12,6)

plt.plot(loss_values, 'b-o')

plt.title("Training loss")
plt.xlabel("Epoch")
plt.ylabel("Loss")

plt.show()
```

## 6) Prediksi dan Evaluasi Model

Tahapan ini adalah bagian dari proses evaluasi dan deployment model, di mana model yang telah dilatih digunakan untuk memprediksi dan mengevaluasi performa pada data uji, serta menyediakan fungsi interaktif untuk penggunaan model pada input baru.

- Prediksi pada Data Uji

Pada Bagian ini, menghasilkan prediksi untuk data uji dan menyimpan logits serta label sebenarnya.

```
print("Predicting labels for {:,} test sentences".format(len(test_input)))

model.eval()

prediction, true_labels = [], []

for batch in test_dataloader:
    batch = tuple(t.to(device) for t in batch)

    b_input_ids, b_input_mask, b_labels = batch

    with torch.no_grad():
        outputs = model(b_input_ids,
                        token_type_ids=None,
                        attention_mask=b_input_mask)

    logits = outputs[0]

    logits = logits.detach().cpu().numpy()
    label_ids = b_labels.to('cpu').numpy()

    prediction.append(logits)
    true_labels.append(label_ids)

print(" DONE.")
```

- Menghitung Matthews Correlation Coefficient (MCC)

Menghitung metrik evaluasi seperti MCC dan akurasi untuk menilai performa model.



```
[ ] from sklearn.metrics import matthews_corrcoef

flat_prediction = [item for sublist in prediction for item in sublist]
flat_prediction = np.argmax(flat_prediction, axis=1).flatten()

flat_true_labels = [item for sublist in true_labels for item in sublist]

mcc = matthews_corrcoef(flat_true_labels, flat_prediction)

print("MCC: %.3f" %mcc)
```

- Menghitung Akurasi

Pada bagian ini, akurasi dihitung menggunakan `accuracy_score` dari `sklearn`.

```
[ ] from sklearn.metrics import accuracy_score

acc = accuracy_score(flat_true_labels, flat_prediction)

print("ACC: %.3f" %acc)
```

- Prediksi Interaktif

Pada bagian ini, prediksi interaktif digunakan untuk mengimplementasikan fungsi untuk tokenisasi, padding, prediksi label judul berita dan mengimplementasikan fungsi interaktif untuk memprediksi dan menampilkan apakah judul berita adalah clickbait atau bukan.

```
[ ] def predict_label(title):
    # Tokenisasi dan padding
    input_ids = tokenizer.encode(title, add_special_tokens=True)
    input_ids = pad_sequences([input_ids], maxlen=MAX_LEN, dtype='long', value=0, truncating='post', padding='post')
    input_ids = torch.tensor(input_ids)

    # Pindahkan ke perangkat CUDA jika tersedia
    if torch.cuda.is_available():
        input_ids = input_ids.to('cuda')
        model.to('cuda')

    # Prediksi label
    with torch.no_grad():
        outputs = model(input_ids)

    logits = outputs.logits
    logits = logits.detach().cpu().numpy()
    predicted_label = logits[0][1]

    return predicted_label

def interactive(title):
    print("Headline:", title)

    # Prediksi label
    label = predict_label(title)
```

b. Scraping Data

Scraping data adalah proses pengambilan data secara otomatis dari situs web untuk keperluan analisis dan pengolahan lebih lanjut. Dalam aplikasi BeBerIn, scraping data digunakan untuk mengumpulkan judul berita dari berbagai sumber online seperti website

GridID dan Detikcom untuk kemudian dianalisis menggunakan model AI IndoBERT guna mendeteksi clickbait. Berikut adalah penjelasan dari masing-masing tahapan:

- Inisialisasi dan Konfigurasi

Mengimpor library yang diperlukan dan menginisialisasi variabel, termasuk URL basis dan rentang tanggal untuk memfilter berita.

```
1 import requests
2 from bs4 import BeautifulSoup
3 import sqlite3
4 import schedule
5 import time
6 from datetime import datetime
7
8 # URLs of the pages to scrape
9 detik_base_url = 'https://www.detik.com/search/searchall?query=berita+hari+ini&siteid=2&sortby=time&result_type=latest'
10 grid_base_url = 'https://www.grid.id/index?page='
11
12 # Define the date range for filtering
13 start_date = datetime.strptime('2024-06-01', '%Y-%m-%d')
14 end_date = datetime.strptime('2024-07-09', '%Y-%m-%d')
15
```

- Scraping Data

Mengambil data berita dari situs Detik.com dan Grid.id, termasuk memeriksa validitas tanggal dan mengumpulkan informasi seperti judul, tautan, dan tanggal.

```
24 # Function to scrape a single page from Detik.com
25 def scrape_detik_page(page_number):
26     url = f'{detik_base_url}{page_number}'
27     response = requests.get(url)
28     if response.status_code != 200:
29         print(f'Failed to retrieve Detik.com page {page_number}. Status code: {response.status_code}')
30         return []
31
32     soup = BeautifulSoup(response.text, 'html.parser')
33     article_items = soup.find_all('article', class_='list-content__item')
34     page_data = []
35
36     for item in article_items:
37         title_tag = item.find('h3', class_='media_title').find('a')
38         link = title_tag['href'] if title_tag and 'href' in title_tag.attrs else None
39         title = title_tag.get_text(strip=True) if title_tag else None
40
41         date_tag = item.find('div', class_='media_date').find('span')
42         date = date_tag.get_text(strip=True).replace(' WIB', '').replace(' detikNews', '').split(' ')[1:4]
43         date = ''.join(date) if date_tag else None
44
45         if is_date_in_range(date):
46             page_data.append({'title': title, 'date': date, 'link': link})
47
48     return page_data
```

```
50 # Function to scrape all pages from Grid.id
51 def scrape_grid_pages():
52     page_number = 1
53     all_data = []
54     while True:
55         url = f'{grid_base_url}{page_number}'
56         response = requests.get(url)
57         if response.status_code != 200:
58             print(f'Failed to retrieve Grid.id page {page_number}. Status code: {response.status_code}')
59             break
60
61         soup = BeautifulSoup(response.content, 'html.parser')
62         articles = soup.find_all('div', class_='news-list__item')
63         if not articles:
64             break
65
66         month_names = {
67             1: 'Januari', 2: 'Februari', 3: 'Maret', 4: 'April',
68             5: 'Mei', 6: 'Juni', 7: 'Juli', 8: 'Agustus',
69             9: 'September', 10: 'Oktober', 11: 'November', 12: 'Desember'
70         }
71         for article in articles:
72             title = article.find('h2', class_='news-list__title').get_text(strip=True) if article.find('h2', class_='n
73             link = article.find('a', class_='news-list__link')['href'] if article.find('a', class_='news-list__link')
```

- Penyimpanan Data

Menyimpan data yang di-scrape ke dalam database SQLite, membuat tabel, dan melakukan commit setelah setiap pengambilan data.

```
91
92 # Function to scrape data from both websites and store in a single SQLite database
93 def scrape_and_store(start_page=1, end_page=5, delay=1):
94     conn = sqlite3.connect('scraped_alldata.db')
95     cursor = conn.cursor()
96
97     # Create tables if not exist
98     cursor.execute('''
99         CREATE TABLE IF NOT EXISTS detik_articles (
100             id INTEGER PRIMARY KEY AUTOINCREMENT,
101             title TEXT,
102             link TEXT,
103             date TEXT
104         )
105     ''')
106     cursor.execute('''
107         CREATE TABLE IF NOT EXISTS grid_articles (
108             id INTEGER PRIMARY KEY AUTOINCREMENT,
109             title TEXT,
110             link TEXT,
111             date TEXT
112         )
113     ''')
```

- Penjadwalan

Menjadwalkan proses scraping untuk berjalan setiap jam dari pukul 23.59 pagi hingga tengah malam, serta sekali pada pukul 00:00, dan menjalankan scheduler secara terus-menerus.

```
146 # Schedule scraping to run from 11 AM to 12 AM
147 for hour in range(23, 24):
148     schedule.every().hour.at(f'{hour:02}:00').do(scrape_and_store)
149
150 # At midnight (00:00)
151 schedule.every().day.at('00:00').do(scrape_and_store)
152
153 # Run the scheduler
154 while True:
155     schedule.run_pending()
156     time.sleep(60) # Wait for 60 seconds
157
```

## Lampiran 4. Sertifikat HKI yang terbit

REPUBLIC INDONESIA KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA	
<b>SURAT PENCATATAN CIPTAAN</b>	
Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:	
Nomor dan tanggal permohonan	: EC00202468067, 18 Juli 2024
<b>Pencipta</b>	
Nama	: <b>Ami Daelami Ichsana, Dwi Intan Af'idah, S.T., M.Kom. dkk</b>
Alamat	: Desa Purwahamba, RT006/RW/006 Kec.Suradadi Kab.Tegal Provinsi Jawa Tengah 52182, Surodadi, Tegal, Jawa Tengah, 52182
Kewarganegaraan	: Indonesia
<b>Pemegang Hak Cipta</b>	
Nama	: <b>Pusat Penelitian dan Pengabdian Masyarakat (P3M) Politeknik Harapan Bersama</b>
Alamat	: Jalan Matarani No. 9, Pesurungan Lor, Kecamatan Margadana 52142, Margadana, Tegal, Jawa Tengah 52142
Kewarganegaraan	: Indonesia
Jenis Ciptaan	: <b>Program Komputer</b>
Judul Ciptaan	: <b>Sistem Deteksi Judul Berita Clickbait Dan Non-Clickbait Di Media Online Dengan Menggunakan Model IndoBert</b>
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia	: 18 Juli 2024, di Tegal
Jangka waktu perlindungan	: Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor pencatatan	: 000643417
adalah benar berdasarkan keterangan yang diberikan oleh Pemohon. Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.	
	a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL ir.6 Direktur Hak Cipta dan Desain Industri
	 IGNATIUS M.T. SILALAH NIP. 196812301996031001
Disclaimer: Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.	

**LAMPIRAN PENCIPTA**

No	Nama	Alamat
1	Ami Daelami Ichsana	Desa Purwahamba, RT006/RW/006 Kec.Suradadi Kab.Tegal Provinsi Jawa Tengah 52182, Surodadi, Tegal
2	Dwi Intan Afidah, S.T., M.Kom.	Desa Grinting RT003/RW001 Kec.Bulakamba Kab.Brebes Provinsi Jawa Tengah 52253, Bulakamba, Brebes
3	Ardi Susanto, M.Cs.	Jl Banyuwangi, RT 08 RW 01, Desa Sumurpanggang Kec.Margadana, Kota Tegal Provinsi Jawa Tengah 52143, Margadana, Tegal



Lampiran 7. Lembar Bimbingan



**SARJANA TERAPAN TEKNIK INFORMATIKA  
POLITEKNIK HARAPAN BERSAMA**

**LEMBAR BIMBINGAN TUGAS AKHIR**

Nama : Ami Daelami Ichsana  
 Nim : 20090133  
 No. Ponsel : 087732886783  
 Judul TA : Komparasi algoritma SVM, LSTM dan IndoBert pada sistem deteksi judul berita clickbait dan non clickbait di media online  
 Dosen Pembimbing II : Ardi Susanto, S.Kom., M.Cs.

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
1.	26-06-2024	Model	sedah baik	
2.	02-07-2024	Aplikasi	tambahkan konta terlewat Jika bukan clickbait.	
3.	21-03-2024	Scraping Data	melanjutkan ke tahapan model	
4.	05-07-2024	model	lanjut ke tahap implementasi	



**SARJANA TERAPAN TEKNIK INFORMATIKA  
POLITEKNIK HARAPAN BERSAMA**

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
5.		Deployment	Website connect ke Database	
6.		Aplikasi	membuat manual book dan dokumen teknis.	
7.	16-07-2024	Manual book & Dokumen teknis	lanjut penyusunan HKI	
8.	17-07-2024	Aplikasi	- Live coding - Lanjut laporan.	



SARJANA TERAPAN TEKNIK INFORMATIKA  
POLITEKNIK HARAPAN BERSAMA

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Ami Daelami Ichsana  
Nim : 20090133  
No. Ponsel : 087732886783  
Judul TA : Komparasi algoritma SVM, LSTM dan IndoBert pada sistem deteksi judul berita clickbait dan non clickbait media online

Dosen Pembimbing I : Dwi Intan Afidah, S.T., M.Kom.

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
1.	21 Maret 2024	Menjelaskan konsep aplikasi yang akan dibangun dengan menunjukkan desain UI/UX	<ul style="list-style-type: none"><li>• Metode yang dimasukkan dalam website cukup satu saja</li><li>• Mencari website yang berisi berita hoax</li><li>• Melakukan scrapping data</li></ul>	
2.	03 April 2024	Menunjukkan hasil scrapping data dari website detikcom dan turnbackhoax	<ul style="list-style-type: none"><li>• Belajar scrapping otomatis data untuk system, agar data terupdate setiap harinya</li><li>• Ganti judul jadi clickbait dan non clickbait</li><li>• Website turnbackhoax sudah fix</li></ul>	
3.	25 April 2024	Menampilkan kodingan scrapping otomatis pada website detikcom	<ul style="list-style-type: none"><li>• Lanjut ke tahapan model</li><li>• Dataset minimal 500</li></ul>	
4.	16 Mei 2024	Menampilkan model	<ul style="list-style-type: none"><li>• SVM(Mechine Learning)</li></ul>	





SARJANA TERAPAN TEKNIK INFORMATIKA  
POLITEKNIK HARAPAN BERSAMA

5.	30 Mei 2024	Menampilkan model	<ul style="list-style-type: none"><li>• LSTM(deep learning)</li><li>• Indobert(Transform)</li><li>• Mencari dataset clickbait dan non-clickbait</li></ul>	
			<ul style="list-style-type: none"><li>• Mencoba memakai BiLSTM Word2vec</li><li>• Untuk model IndoBert di tuning kembali</li><li>• Scrapping otomatis 3 website</li></ul>	
6.	28 Juni 2024	Deployment aplikasi	<ul style="list-style-type: none"><li>• Membuat website dashboard 3 menu : GridID, Detikcom, dan report.</li><li>• Website Connect to database</li></ul>	
7.	05 Juli 2024	Website	<ul style="list-style-type: none"><li>• Perbaikan proses scraping scheduling</li><li>• Buat fitur upload file excel</li><li>• Manualbook</li></ul>	
8.	8 Juli 2024	Website	<ul style="list-style-type: none"><li>• Tambah menu report di website yang terhubung di menu report streamlit</li><li>• Tambah fitur download setelah upload file excel</li></ul>	



**SARJANA TERAPAN TEKNIK INFORMATIKA  
POLITEKNIK HARAPAN BERSAMA**

9.	24 Juli 2024	Streamlit dan Laporan	<ul style="list-style-type: none"><li>• Scrapingan dari bulan januari-juli</li><li>• Revisi Laporan Bab I</li></ul>	
10.	25 Juli 2024	Streamlit dan Laporan	<ul style="list-style-type: none"><li>• Tabel data dibuat perpage</li><li>• Dibuat menu report yang memuat all data dan per bulan</li></ul>	

Tegal, ..... Juli ..... 2024

Dosen Pembimbing I

Dwi Intan Af'idah, S.T., M.Kom.

NIPY. 03.021.490