

Implementasi Deteksi Judul Berita *Clickbait* Berbahasa Indonesia dengan *pre-trained model* Multilingual BERT Pada Aplikasi Berbasis *Chrome Extension*

Girinoto¹⁾, Dhana Arvina Alwan²⁾, Gusti Agung Ngurah Gde K.T. D.³⁾, Olga Geby Nabila⁴⁾, Arizal⁵⁾, Dimas Febriyan Priambodo^{6*)}

^{1, 2, 3, 4)} Rekayasa Perangkat Lunak Kriptografi, Politeknik Siber dan Sandi Negara

⁵⁾ Rekayasa Perangkat Keras Kriptografi, Politeknik Siber dan Sandi Negara

⁶⁾ Rekayasa Keamanan Siber, Politeknik Siber dan Sandi Negara

¹⁾ girinoto@poltekssn.ac.id, ²⁾ dhana.arvina@student.poltekssn.ac.id, ³⁾ gusti.agung@student.poltekssn.ac.id,

⁴⁾ olga.geby@student.poltekssn.ac.id, ⁵⁾ arizal@poltekssn.ac.id, ^{6*)} dimas.febriyan@poltekssn.ac.id.

ABSTRACT

Clickbait news title is often used by online news portal. The purpose of clickbait is to attract reader to open and read the news. Furthermore, news containing clickbait title can give negative impact by reducing the essence of important news. Therefore, clickbait detection tool is needed to avoid the clickbait news title. Chrome extension was chosen in this study because it supports all Chrome based browsers, such as Google Chrome, Chromium, Microsoft Edge, and Opera so that many users apply this program. In this study, Chrome extension-based application was designed and integrated by using artificial intelligence model. This application also utilized the availability of pre-trained multilingual BERT model as Natural Language Processing (NLP) which will be used to predict a clickbait news title. This study used Multilingual BERT model as NLP because this model has been trained into 104 languages, including Bahasa Indonesia and it has significant performance. The result of this study can detect clickbait news along with 92% of AUC-ROC value.

Keywords: Chrome Extension, Clickbait, BERT, Natural Language Processing, Transformers.

I. PENDAHULUAN

Clickbait merupakan sebuah upaya dari penulis untuk membuat judul yang mampu menarik perhatian atau emosional pembaca. Hal ini terjadi karena adanya *information gap* antara hal-hal yang diketahui dan hal-hal yang ingin diketahui. Loewenstein (1994) mengemukakan Teori *Information Gap* yang menyatakan bahwa setiap kali kita melihat celah “antara apa yang kita ketahui dan apa yang ingin kita ketahui,” maka akan menimbulkan sebuah emosionalitas yang menghasilkan perasaan kehilangan sesuatu. Individu yang ingin tahu termotivasi untuk mendapatkan informasi yang hilang tersebut. Sehingga pembaca akan cenderung untuk melihat atau membuka informasi dengan judul *clickbait* (Hadiyat, 2019).

Clickbait menjadi semakin banyak dijumpai, salah satunya disebabkan oleh sistem pengiklanan yang banyak ditentukan dengan banyaknya klik pada website. Semakin banyak klik yang dilakukan pengunjung, semakin tinggi juga popularitas suatu website (Chen et al., 2015). *Clickbait* populer digunakan sebagai jalan pintas untuk menghasilkan uang, hal itu juga yang menyebabkan maraknya *clickbait* ini.

Judul berita yang mengandung unsur *clickbait* sering menggunakan kata kunci yang sedang tren namun isi berita tersebut tidak mengungkapkan informasi apapun, hanya berisikan pesan kontroversial. *Clickbait* memanfaatkan sisi psikologis dari seorang manusia yaitu rasa keingintahuan manusia. Rasa ingin tahu seperti itu muncul ketika manusia ingin tahu tentang sesuatu yang baru. Celah rasa ingin tahu tersebut dapat mudah dimanfaatkan dengan memberikan pesan-pesan dengan kalimat yang mempengaruhi pembaca dalam headline *clickbait* tentang informasi baru atau kontroversial, memancing rasa ingin tahu pembaca, dan mengarahkan mereka untuk mengklik judul tersebut (Fakhruzzaman & Gunawan, 2021).

Banyak peneliti yang telah membuat metode deteksi *clickbait* menggunakan *Artificial Intelligence*. Tetapi sebagian besar hanya berupa teori, dan masih sedikit yang mengembangkan implementasi berupa alat (Fakhruzzaman & Gunawan, 2021). Didukung dengan masyarakat Indonesia yang lebih mudah terkecoh dengan *clickbait* yang ditunjukkan dengan artikel atau konten dengan judul yang memiliki unsur *clickbait* memiliki rata - rata *traffic* yang lebih tinggi daripada artikel dengan judul yang bukan *clickbait*. Hal ini menjadi salah satu hal yang mendasari dibuatnya aplikasi ini.

II. TINJAUAN PUSTAKA

Penelitian pendahulu dari Agus Purwo (Handoko, 2006) (Handoko, 2007) membuat aplikasi pengolah bahasa secara *dedicated* dan pada update terakhir memantik peneliti untuk mengikuti trend terkini dengan mengembangkan sebuah aplikasi berbasis web yang mengimplementasikan metode deteksi *clickbait* menggunakan INDOBERT (Fakhruzzaman & Gunawan, 2021). Aplikasi berbasis web ini dirasa kurang efisien. Hal ini karena pengguna harus membuka halaman web terlebih dahulu untuk mengetahui apakah suatu judul artikel tergolong ke dalam *clickbait* atau tidak. Hal tersebut dapat dibenahi dengan merubah aplikasi yang awalnya berbasis web menjadi berbasis browser extension. Dengan *browser extension*, pengguna dapat lebih mudah menelusuri apakah suatu kalimat/judul artikel termasuk ke dalam *clickbait* tanpa perlu membuka halaman baru dan akses yang lebih cepat dan praktis (Botnevik et al., 2020).

Penggunaan *pre-trained* model Multilingual BERT didasari oleh penelitian sebelumnya yang membandingkan nilai AUC-ROC pada model ini yang didapati lebih baik yakni 0.92 jika dibandingkan dengan model INDOBERT yang memiliki nilai AUC-ROC 0.89 dengan proses evaluasi pada dataset yang sama (Fakhruzzaman & Gunawan, 2021). Pada penelitian sebelumnya, penulis lebih memilih menggunakan model INDOBERT dan mengorbankan akurasi model hingga 3% dikarenakan keterbatasan resource yang ada yaitu berupa keterbatasan memori yang mempengaruhi biaya server dan hosting pada pembangunan web (Fakhruzzaman & Gunawan, 2021). Oleh karena itu, pada penelitian ini akan digunakan model Multilingual BERT untuk judul artikel yang memiliki unsur *clickbait* berbahasa Indonesia dikarenakan tingginya akurasi dari model ini dengan mengesampingkan kebutuhan resource tinggi untuk menjalankan model ini.

2.1 Multilingual BERT

BERT yang memiliki kepanjangan Bidirectional Encoder Representation from Transformer adalah sebuah teknologi deep learning yang dikembangkan untuk memecahkan Natural Language Processing (Devlin et al., 2018). Model BERT telah mendapatkan nilai state-of-the-art baru dalam 11 Natural Language Processing *task* , salah satunya adalah GLUE, MultiNLI dan SQUAD (Devlin et al., 2018).

Multilingual BERT dengan kata lain adalah versi BERT namun dengan multi-bahasa, merupakan salah satu *pre-trained* model BERT namun yang membedakan Multilingual BERT dan model lainnya adalah model ini dilatih dengan 104 bahasa termasuk bahasa indonesia (Pires et al., 2019). Model ini sangat berguna untuk memecahkan masalah yang mencakup banyak bahasa. Dibandingkan dengan sesamanya seperti model UDify dan Ing-free , model mbert base unggul dalam mengidentifikasi bahasa (Pires et al., 2019).

Tabel 1. Akurasi Identifikasi bahasa (Pires et al., 2019).

	mBERT	UDify	Ing-free
[cls]	.935	.938	.796
[cls] , cent.	.867	.851	.337
mean-pool	.919	.896	.230
Mean-pool, cent	.285	.243	.247

Digunakannya Multilingual BERT pada penelitian ini, selain dari telah dilatih dengan sejumlah kata berbahasa indonesia, model ini mengungguli *pre-trained* Model yang dibuat khusus untuk bahasa indonesia yaitu INDOBERT [8]. Hal ini merupakan hal yang mengejutkan dikarenakan model yang telah dilatih dengan 220 juta kata dari wikipedia, artikel berita dan korpus web indonesia [8] belum dapat mengungguli Multilingual BERT yang notabennya dilatih dengan banyak bahasa dan tidak terfokus hanya bahasa Indonesia.

2.2 Dataset

Dalam melatih model nantinya, tentu saja akan membutuhkan sebuah *dataset* untuk melakukan untuk melakukan *training* dan modeling. Studi ini menggunakan *dataset* judul berita yang diambil dari 12 *website* berita yang dapat dilihat pada Tabel 2 dan proses labeling dilakukan oleh 3 orang mahasiswa (William & Sari, 2020). Judul berita berbahasa indonesia dari 9 kategori berita yang sudah teranotasi dengan rincian 5297 baris data berlabel *non-clickbait* dan 3316 baris data berlabel *clickbait*. Dataset ini memiliki nilai Fleiss' Kappa Interrater agreement 0,42 (William & Sari, 2020).

Tabel 2. Sumber Website *dataset*

No.	Website sumber
1	www.detik.com
2	www.fimela.com
3	www.kapanlagi.com
4	www.kompas.com
5	www.liputan6.com
6	www.okezone.com
7	www.posmetro-medan.com
8	www.republika.co.id
9	www.sindonews.com
10	www.sindonews.com
11	www.tribunnews.com
12	www.wowkeren.com

Selain *dataset* yang telah dianotasi, *dataset* ini dipilih karena memiliki kredibilitas dikarenakan cukup sering digunakan oleh peneliti - peneliti lainnya untuk riset tentang judul berita *clickbait* berbahasa Indonesia.

Dikarenakan jumlah label judul berita yang tidak seimbang, maka dilakukan *undersampling* pada dataset untuk menyeimbangkan jumlah label *clickbait* dan *non-clickbait*. Setelah dilakukan *undersampling* Jumlah data yang digunakan untuk proses training adalah 6632 judul berita dengan label yang sama banyak dan seimbang. Hal ini dilakukan untuk menghindari *overfit* atau *underfit* dari model nantinya yang disebabkan oleh tidak seimbangnya data *training*.

2.3 REST API

API (Application Programming Interface) bertujuan untuk menghubungkan dua program untuk saling berkomunikasi. Client program akan menggunakan API untuk melakukan komunikasi kepada server. Secara umum, API akan mengekspos data dan fungsi untuk melakukan pertukaran informasi antar program komputer (Masse, 2011).

Gaya arsitektur “Representational State Transfer” (REST) umumnya diterapkan pada desain API untuk layanan web modern. REST API dirancang untuk memenuhi kebutuhan client program. Pada REST API, JSON merupakan format yang sering digunakan. Pada pertukaran data, penggunaan format JSON sangatlah ringan dan sederhana. JSON menggunakan beberapa bagian pada pemrograman Javascript, karenanya dibutuhkan browser-based JavaScript clients (Masse, 2011).

Penggunaan REST API pada penelitian ini adalah untuk menghubungkan frontend dan backend pada arsitektur aplikasi ini yang mana nantinya API akan menerima input dari browser pada sisi user dan memberikan masukan ke model untuk diprediksi.

2.4 Browser Extension

Web browser yang modern sudah mendukung adanya arsitektur yang memungkinkan pihak ketiga untuk meningkatkan atau memodifikasi fungsionalitas dari web browser dalam banyak cara melalui sebuah browser extension (Dhawan & Ganapathy, 2009).

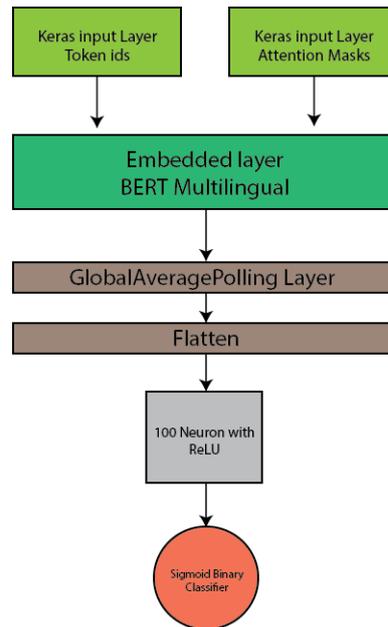
Pada penelitian ini, kami mengembangkan browser extension pada Chrome. Ketika pengguna menandai teks menggunakan highlight, kemudian mengklik browser extension, modul JavaScript akan digunakan untuk mengambil teks yang diseleksi dan mengirim teks tersebut ke API. Kemudian API yang berisi fungsi-fungsi akan mengembalikan hasil ke web browser kembali. Ketika hasilnya dikembalikan dari server, modul JavaScript lain dipanggil untuk menampilkan hasilnya.

Penulisan JavaScript pada browser extension akan memberikan banyak keuntungan salah satunya adalah bahwa JavaScript akan relevan dan relatif stabil dengan perkembangan browser di masa mendatang. Selain itu, JavaScript juga mudah dipindahkan ke seluruh platform dan versi browser (Dhawan & Ganapathy, 2009).

III. METODE PENELITIAN

3.1 Modelling

Permodelan menggunakan konfigurasi neural network yang diilustrasikan pada Gambar 1. Konfigurasi ini diawali dengan 2 Keras Input layer yang masing masing layer-nya mengolah id token dan *attention masks* yang selanjutnya baru di teruskan ke Multilingual BERT layer (Fakhruzzaman & Gunawan, 2021).



Gambar 1. Konfigurasi *Neural Network*

Hasil dari layer Multilingual BERT selanjutnya diteruskan dense layer GlobalAveragePolling dan Flatten layer (Fakhruzzaman & Gunawan, 2021). Terakhir data diteruskan dengan ReLU dan diklasifikasikan menggunakan fungsi sigmoid yang menghasilkan label *clickbait* atau *non clickbait* (Fakhruzzaman et al., 2021).

Proses *training* model yang akan digunakan untuk aplikasi dijalankan dengan 4 *epoch* dan ukuran *batch* sebesar 16, Sedangkan untuk proses evaluasi dilakukan sebanyak 5 kali yang nantinya nilai AUC-ROC akan didapat dari rata-rata dari kelima proses training tersebut (Fakhruzzaman & Gunawan, 2021). AUC-ROC *curve* adalah sebuah pengukuran performa untuk sebuah permasalahan yang berhubungan dengan klasifikasi (Narkhede, 2018). ROC (*Receiver Operating Characteristics*) adalah kurva probabilitas dan AUC (*Area Under The Curve*) adalah area yang menunjukkan tingkat keakuratan model dalam memberikan suatu label (Narkhede, 2018). Semakin tinggi nilai AUC maka semakin tinggi ketepatan model tersebut akan menilai suatu label dengan benar dan akurat. Kurva ROC dapat dihitung menggunakan TPR dan FPR dengan persamaan sebagai berikut.

$$TPR (True Positive Rate) = \frac{TP}{TP + FN}$$
$$FPR (False Positive Rate) = \frac{FP}{TN + FP}$$

Dengan TP (*True Positive*), TN (*True Negative*), FP (*False Positive*) dan FN (*False Negative*).

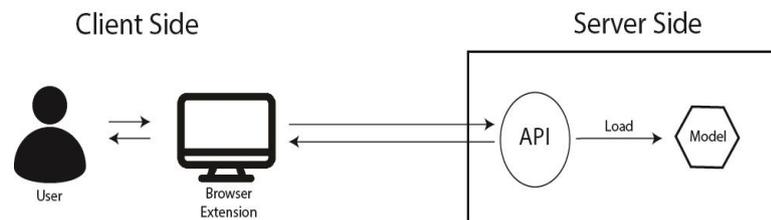
Nantinya *pre-trained* model yang digunakan yakni Multilingual BERT *uncased* dan Multilingual BERT *cased* akan dibandingkan untuk melihat yang mana memiliki nilai evaluasi yang lebih baik, sehingga dapat diimplementasikan sebagai model prediksi aplikasi nantinya. Dengan perbedaan dataset *training* untuk kedua model tersebut hanya pada bagian tidak adanya fungsi *lowercase* untuk mengubah dataset menjadi huruf kecil pada data training untuk model M-BERT *cased*.

3.2 Aplikasi

Pembangunan aplikasi ini menggunakan REST API sebagai penghubung antara aplikasi *client side* yaitu *browser extension* dan model yang sudah dilatih. Gambar 2 menunjukkan arsitektur *data flow* dari user ke model yang dihubungkan dengan API yang

menerima data dengan format JSON yang nantinya akan diprediksi oleh model dan mengembalikan hasil ke user berupa data label dan probabilitas dari prediksi.

Proses percobaan aplikasi ini masih menggunakan *localhost* yang menggunakan library ngrok untuk melakukan *tunneling* localhost ke internet, sehingga dapat diakses dari jaringan internet. *Resource* yang digunakan merupakan *resource* dari Google Colaboratory sehingga memiliki kemampuan yang cukup mumpuni untuk melakukan latihan model dan kecepatan internet yang stabil.



Gambar 2. Data flow Aplikasi

Pembangunan API menggunakan library python fastAPI. Penggunaan library fastAPI dikarenakan cara penggunaan yang cukup sederhana sehingga dapat mempercepat proses pembangunan. Namun untuk *hosting* disini masih menggunakan server ngrok dikarenakan keterbatasan sumber daya yang ada.

Selanjutnya dalam proses pengembangan extension untuk browser berbasis chrome, dalam penelitian ini percobaan dilakukan pada *browser* Google Chrome. Pengembangan ini menggunakan JavaScript, HTML, dan CSS. Hal utama yang dilakukan oleh ekstensi ini adalah mempersingkat akses user terhadap model deteksi, hanya dengan menandai judul berita yang ingin dideteksi dan di salin ke dalam ekstensi tanpa harus membuka tab baru ataupun laman web baru.

Terdapat empat komponen dalam membangun ekstensi ini. Diantaranya file *manifest.json*, halaman popup, Content Script, dan Events Page. File utama yang dibutuhkan dalam membangun chrome extension adalah file “manifest.json”. File ini merupakan metadata yang disajikan dalam format JSON yang memberikan informasi tentang ekstensi pada Chrome, seperti nama, versi, deskripsi, browser action, content script, background, icon, dan izin yang diperlukan.

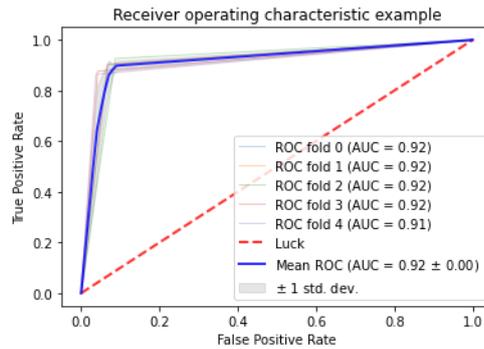
Ekstensi yang dibangun ini memanfaatkan halaman popup sebagai tampilan utamanya, jadi ketika pengguna mengklik ekstensi, akan muncul halaman popup. Halaman popup dibangun menggunakan standar bahasa markup yaitu HTML. Pada halaman ini akan ditampung data berupa string yang diinputkan pengguna, kemudian jika pengguna menekan tombol “Check”, maka inputan tersebut akan dikirimkan ke server untuk diprediksi apakah tergolong ke dalam clickbait.

Untuk mendapatkan teks yang diseleksi oleh pengguna, aplikasi ekstensi ini memanfaatkan content script. Selanjutnya untuk mendapatkan hasil prediksi suatu headline berita, kami memanfaatkan notification API pada chrome, yang mana akan menampilkan sebuah notifikasi saat mendapatkan respon dari server.

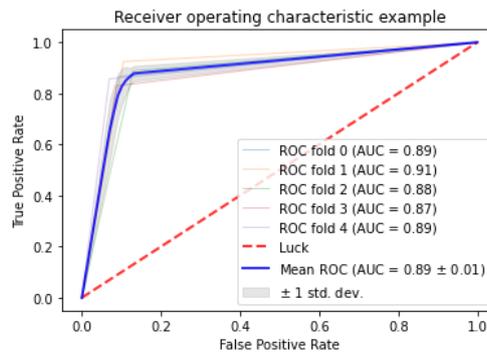
Metode penelitian merupakan prosedur dan teknik penelitian. Antara satu penelitian dan penelitian lain, prosedur dan tekniknya akan berbeda. Kalau tidak berbeda berarti penelitian itu hanya mengulang penelitian yang telah ada sebelumnya.

IV. HASIL DAN PEMBAHASAN

Dari proses evaluasi model dengan model *pre-trained* Multilingual BERT *uncased* dan Multilingual BERT *cased*, dari 5 kali percobaan *training* didapatkan rata - rata AUC-ROC bernilai 92% pada M-BERT *uncased* dan nilai 89% pada model M-BERT *cased*.

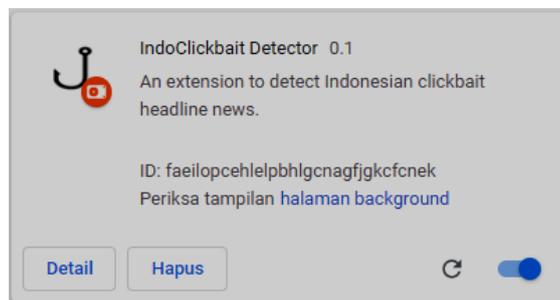


Gambar 3. Kurva ROC evaluasi model M-BERT *uncased*

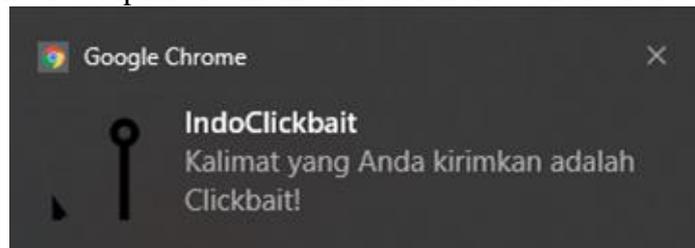


Gambar 4. Kurva ROC evaluasi model M-BERT *cased*

Dari kedua gambar diatas dapat dilihat bahwa model yang menggunakan M-BERT *uncased* memiliki nilai AUC-ROC lebih tinggi 3% dibandingkan model yang menggunakan M-BERT *cased*, sehingga implementasi model ini ke aplikasi akan menggunakan M-BERT *uncased* dikarenakan selain performa yang lebih unggul juga judul - judul berita yang sering ditemukan di internet dituliskan menggunakan huruf kapital semua yang mana itu akan memiliki hasil yang berbeda jika menggunakan model M-BERT *cased*.



Gambar 5. Aplikasi Chrome Extension IndoClickbait Detector



Gambar 6. Notifikasi hasil deteksi *headline* berita.

Selanjutnya pada pembangunan aplikasi ekstensi chrome, aplikasi yang dibangun di uji pada *browser* Google chrome. Pengguna dapat menggunakan aplikasi ini dengan menginstall aplikasi seperti yang ditampilkan pada Gambar 5 , memilih sebuah judul berita

di internet , lalu salin judul tersebut ke ekstensi, yang nantinya hasil dari prediksi akan ditampilkan dalam bentuk jendela notifikasi *pop up* pada Gambar 6.

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari penelitian yang dilakukan dapat disimpulkan Model Multilingual BERT uncased memiliki keunggulan dibandingkan model cased dalam tugas untuk mengklasifikasikan judul berita *clickbait* dengan dataset pada penelitian ini, yang memiliki nilai AUC-ROC sebesar 92%. Sehingga model Multilingual BERT uncased digunakan sebagai model prediksi untuk implementasi pada aplikasi chrome extension. Pembangunan aplikasi chrome extension ini juga menggunakan arsitektur REST API dalam bertukar informasi antara model prediksi dan aplikasi sehingga implementasi ini masih bisa dikembangkan untuk penelitian selanjutnya.

5.2 Saran

Untuk penelitian berikutnya disusun menggunakan metode modeling yang lain sehingga dapat dibandingkan atau dapat dicari gap analisis yang terjadi.

DAFTAR PUSTAKA

- Botnevik, B., Sakariassen, E., & Setty, V. (2020). (2020). Brenda: Browser extension for fake news detection. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2117–2120.
- Chen, Y., Conroy, N. K., & Rubin, V. L. (2015). News in an online world: The need for an “automatic crap detector.” *Proceedings of the Association for Information Science and Technology*, 1–4.
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). *Bert: Pre-training of deep bidirectional transformers for language understanding*. *arXiv*.
- Dhawan, M., & Ganapathy, V. (2009). Analyzing Information Flow in JavaScript-Based Browser Extensions. *2009 Annual Computer Security Applications Conference*, 382–391. <https://doi.org/10.1109/ACSAC.2009.43>
- Fakhruzzaman, M. N., & Gunawan, S. W. (2021). *Web-based Application for Detecting Indonesian Clickbait Headlines using IndoBERT*. *arXiv*.
- Fakhruzzaman, M. N., Jannah, S. Z., Ningrum, R. A., & Fahmiyah, I. (2021). *Clickbait Headline Detection in Indonesian News Sites using Multilingual Bidirectional Encoder Representations from Transformers (M-BERT)*. *arXiv*.
- Hadiyat, Y. D. (2019). Clickbait on Indonesia Online Media. *Journal Pekommas*, 4(1), 1. <https://doi.org/10.30818/jpkm.2019.2040101>
- Handoko, A. P. (2006). Aplikasi Pengolah Bahasa Alami Untuk Operasi Queri Database. *Jurnal Ilmiah Sinus*, 4(2). <https://doi.org/http://dx.doi.org/10.30646/sinus.v4i2>
- Handoko, A. P. (2007). Aplikasi Pengolah Bahasa Alami Untuk Operasi Boolean Antar Citra. *Jurnal Ilmiah Sinus*, 5, 2. <https://doi.org/http://dx.doi.org/10.30646/sinus.v5i2>
- Masse, M. (2011). *REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces*. O’Reilly Media, Inc.
- Narkhede, S. (2018). *Understanding AUC - ROC Curve | by Sarang Narkhede | Towards Data Science*. Towards Data Science. <https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>
- Pires, T., Schlinger, E., & Garrette, D. (2019). How multilingual is multilingual BERT? *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4996--5001. <https://doi.org/10.18653/v1/P19-1493>
- William, A., & Sari, Y. (2020). CLICK-ID: A novel dataset for Indonesian clickbait headlines. *Data in Brief*, 32(10623), 1.