# IMPLEMENTASI *CONDITIONAL GENERATIVE ADVERSARIAL NETWORK* (cGAN) DAN *BIDIRECTIONAL ENCODER REPRESENTATIONS FROM TRANSFORMERS* (BERT) UNTUK KLASIFIKASI BIDANG ILMU ARTIKEL ILMIAH TERINDEKS GARUDA

**LAPORAN TUGAS AKHIR**

Laporan ini disusun untuk memenuhi salah satu syarat untuk menyelesaikan program studi Teknik Informatika S-1 pada Fakultas Teknologi Industri Universitas Islam Sultan Agung


**Disusun Oleh:**

**AINUN DEA RAHAYU**

**NIM 32602100017**


**FAKULTAS TEKNOLOGI INDUSTRI**

**UNIVERSITAS ISLAM SULTAN AGUNG**

**SEMARANG**

**2025**

# IMPLEMENTASI *CONDITIONAL GENERATIVE ADVERSARIAL NETWORK* (cGAN) DAN *BIDIRECTIONAL ENCODER REPRESENTATIONS FROM TRANSFORMERS* (BERT) UNTUK KLASIFIKASI BIDANG ILMU ARTIKEL ILMIAH TERINDEKS GARUDA

***FINAL PROJECT***

*This report was prepared to fulfill one of the requirements for completing the Undergraduate Informatics Engineering study program at the Faculty of Industrial Technology, Sultan Agung Islamic University.*

***Arranged by:***

**AINUN DEA RAHAYU**

**NIM 32602100017**

***MAJORING OF INFORMATICS ENGINEERING***

***FACULTY OF INDUSTRIAL TECHNOLOGY***

***SULTAN AGUNG ISLAMIC UNIVERSITY***

**SEMARANG**

**2025**

**LEMBAR PENGESAHAN**
**TUGAS AKHIR**

# IMPLEMENTASI *CONDITIONAL GENERATIVE ADVERSARIAL NETWORK* (cGAN) DAN *BIDIRECTIONAL ENCODER REPRESENTATIONS FROM TRANSFORMERS* (BERT) UNTUK KLASIFIKASI BIDANG ILMU ARTIKEL ILMIAH TERINDEKS GARUDA

**AINUN DEA RAHAYU**
**NIM 32602100017**

Telah dipertahankan di depan tim penguji seminar tugas akhir
Program Studi Teknik Informatika
Universitas Islam Sultan Agung
Pada tanggal : 16 Juli 2025

**TIM PENGUJI SEMINAR TUGAS AKHIR:**

| | | |
|---|---|---|
| **Andi Riansyah, ST, M. Kom**<br>NIDN. 0609108802<br>(Ketua Penguji) | ..... | 6 Agustus 2025 |
| **Bagus Satrio W. P., S.Kom., M.Cs**<br>NIDN. 1027118801<br>(Anggota Penguji) | ..... | 6 Agustus 2025 |
| **Badie'ah, ST., M.Kom**<br>NIDN. 0619018701<br>(Pembimbing) | ..... | 06 Agustus 2025 |

Semarang, 06 Agustus 2025

Mengetahui,

Kaprodi Teknik Informatika
Universitas Islam Sultan Agung

**Moch. Taufik, S.T., M.IT**
NIDN. 0622037502

## SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama : Ainun Dea Rahayu

NIM : 32602100017

Judul Tugas Akhir : Implementasi *Conditional Generative Adversarial Network* (cGAN) dan *Bidirectional Encoder Representations From Transformers* (BERT) untuk Klasifikasi Bidang Ilmu Artikel Ilmiah Terindeks Garuda

Dengan bahwa ini saya menyatakan bahwa judul dan isi Tugas Akhir yang saya buat dalam rangka menyelesaikan Pendidikan Strata Satu (S1) Teknik Informatika tersebut adalah asli dan belum pernah diangkat, ditulis ataupun dipublikasikan oleh siapa pun baik keseluruhan maupun sebagian, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apabila di kemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut pernah diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang, 06 Agustus 2025

Yang Menyatakan


Ainun Dea Rahayu

## PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini :

| | |
|---|---|
| Nama | : Ainun Dea Rahayu |
| NIM | : 32602100017 |
| Program Studi | : Teknik Informatika |
| Fakultas | : Teknologi industri |

Dengan ini menyatakan Karya Ilmiah berupa Tugas akhir dengan Judul : Implementasi *Conditional Generative Adversarial Network* (cGAN) dan *Bidirectional Encoder Representations From Transformers* (BERT) untuk Klasifikasi Bidang Ilmu Artikel Ilmiah Terindeks Garuda

Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan Hak bebas Royalti Non-Eksklusif untuk disimpan, dialihmediakan, dikelola dan pangkalan data dan dipublikasikan diinternet dan media lain untuk kepentingan akademis selama tetap menyantumkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran Hak Cipta/Plagiarisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan Agung.

Semarang, 6 Agustus 2025

Yang menyatakan,


Ainun Dea Rahayu

# KATA PENGANTAR

Segala puji dan syukur saya panjatkan ke hadirat Allah SWT atas limpahan rahmat dan karunia-Nya, sehingga saya dapat menyelesaikan Tugas Akhir ini dengan judul "Implementasi *Conditional Generative Adversarial Network* (cGAN) dan *Bidirectional Encoder Representations From Transformers* (BERT) untuk Klasifikasi Bidang Ilmu Artikel Ilmiah Terindeks Garuda" Tugas Akhir ini disusun sebagai salah satu syarat menyelesaikan studi serta dalam rangka memperoleh gelar sarjana (S-1) pada program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Islam Sultan Agung.

Tugas Akhir ini disusun dan dibuat dengan adanya bantuan dari berbagai pihak, materi maupun teknis, oleh karena itu saya selaku penulis mengucapkan terima kasih kepada:

1. Rektor UNISSULA Bapak Prof. Dr. H. Gunarto, S.H., M.H yang mengizinkan penulis menimba ilmu dikampus ini.
2. Dekan Fakultas Teknologi Industri Ibu Dr. Ir. Novi Marlyana, S.T., M.T., IPU., ASEAN. Eng.
3. Dosen Pembimbing penulis Ibu Badie'ah, ST., M.Kom yang telah meluangkan waktu, membimbing dan memberikan banyak nasehat dan saran.
4. Orang tua penulis, Bapak Rukiman dan Ibu Musrifah, yang selalu memberikan restu, dukungan, serta Doa dalam menyelesaikan Tugas Akhir ini.
5. Saudara kandung saya satu-satunya yaitu Bima Ramadhan dan teman-teman atas bantuan, semangat, dan dukungannya.

Dengan rendah hati, penulis menyadari bahwa laporan masih memiliki banyak kekurangan dalam kualitas dan isi. karena itu, penulis mengharapkan kritikan dan saran yang membangun untuk penyempurnaan di masa depan.

Semarang, 06 Agustus 2025

Ainun Dea Rahayu

# DAFTAR ISI

## DAFTAR GAMBAR

# DAFTAR TABEL

## ABSTRAK

Jumlah publikasi karya ilmiah terus meningkat 2,6 hingga 3 juta artikel per tahun, yang menciptakan tantangan besar dalam proses otomatisasi pengelompokan dan klasifikasi bidang ilmu. Tantangan ini semakin kompleks ketika distribusi data antar kelas tidak seimbang. Untuk mengatasi hal ini, penelitian ini menggabungkan dua pendekatan yaitu *Conditional Generative Adversarial Network* (cGAN) dan *Bidirectional Encoder Representations from Transformers* (BERT). Model cGAN dimanfaatkan untuk menciptakan data sintetis bagi kelas minoritas, sementara BERT digunakan untuk klasifikasi berbasis pemahaman konteks. Data hasil sintesis kemudian digabung dengan data asli dan diuji menggunakan metrik seperti akurasi, presisi, *recall*, dan F1-*score*. Hasilnya, metode ini mampu mencapai akurasi hingga 87% serta memperbaiki keseimbangan distribusi prediksi antar bidang ilmu. Pendekatan cGAN-BERT ini diharapkan efektif dalam mengatasi masalah data tidak seimbang dan memiliki potensi besar untuk diterapkan pada sistem klasifikasi otomatis artikel ilmiah.


Kata Kunci : Augmentasi Data, cGAN, BERT, Klasifikasi bidang ilmu, Artikel ilmiah



## *ABSTRACT*

*The number of scientific publications continues to grow, reaching approximately* 2.6 *to* 3 *million articles per year, which presents a major challenge in automating the grouping and classification of scientific fields. This challenge becomes even more complex when the data distribution across classes is imbalanced. To address this issue, this study integrates two approaches: Conditional Generative Adversarial Network* (cGAN) *and Bidirectional Encoder Representations from Transformers* (BERT). *The cGAN model is employed to generate synthetic data for underrepresented classes, while* BERT *serves as a context-aware classification model. The synthetic data is then combined with the original dataset and evaluated using metrics such as accuracy, precision, recall, and* F1-*score. The results show that this method achieves an accuracy of up to* 87% *and improves the balance of prediction distribution across scientific fields. The* cGAN-BERT *approach has proven effective in addressing data imbalance issues and holds significant potential for application in automatic classification systems for scientific articles.*


*Keywords: Data Augmentation,* cGAN, BERT, *Scientific Field Classification, Scholarly Articles*

# BAB I
# PENDAHULUAN

## 1.1 Latar Belakang

Dalam era digitalisasi ini, jumlah publikasi karya ilmiah terus meningkat 2,6 hingga 3 juta artikel per tahun, hal ini menciptakan tantangan bagi peneliti ketika mencari jurnal yang relevan dengan bidang kajian mereka. Kecepatan dan kemudahan dalam mendapatkan data terkini untuk penelitian juga dipengaruhi oleh kemajuan teknologi (Dameani 2021). Kecerdasan buatan (*Artificial Intelligence*/AI) telah membuat dampak signifikan untuk berbagai bidang, termasuk dalam sistem klasifikasi (Hajkowicz dkk., 2023). Dalam dunia akademik, sistem klasifikasi memiliki peran penting dalam membantu peneliti untuk menemukan publikasi ilmiah yang relevan dengan penelitian yang sedang dilakukan (Li dkk., 2024).

Salah satu *platform* yang digunakan untuk mengakses publikasi akademik di Indonesia adalah Garba Rujukan Digital (Garuda). *Platform* ini menyediakan berbagai jurnal yang berisi artikel-artikel baik ilmiah maupun non ilmiah yang bersumber dari berbagai peneliti yang mempunyai tingkat akreditasi rendah maupun tinggi serta jurnal-jurnal internasional terpublikasi (Sa'adah 2022). Garuda mengindeks ribuan jurnal dari berbagai disiplin ilmu. Namun, sistem pencarian yang ada masih memiliki keterbatasan dalam memahami konteks semantik dari artikel yang dicari oleh pengguna sehingga relevansi konten masih harus ditelusuri lebih mendalam agar mendapatkan informasi yang diharapkan.

Keterbatasan dalam memahami hubungan semantik antar artikel menjadi masalah utama dalam sistem pengklasifikasian jurnal terpublikasi, adanya ketidakseimbangan jumlah publikasi dalam berbagai bidang penelitian juga membuat beberapa subjek lebih mudah ditemukan dibandingkan subjek yang lain (Hafiz dan Sudarmilah 2023). Untuk mengatasi permasalahan tersebut, diperlukan pendekatan berbasis pemodelan bahasa alami *Natural Language Processing* (NLP) untuk membantu meningkatkan akurasi sistem klasifikasi jurnal terpublikasi.

Metode yang dapat digunakan untuk permasalahan dalam penelitian ini adalah *Conditional Generative Adversarial Network* (cGAN). cGAN adalah arsitektur pembelajaran mesin yang melibatkan dua jaringan saraf yang disebut *generator* dan *discriminator* (Bhat dan Nanjundegowda 2025). *Generator* berupaya menghasilkan data baru sehingga dapat serupa dengan data asli, sedangkan *discriminator* mengevaluasi keaslian data tersebut. Interaksi keduanya menciptakan kompetisi, memungkinkan *generator* menghasilkan data yang semakin realistis. cGAN membantu meningkatkan kinerja model, terutama jika dataset awal memiliki keterbatasan atau ketidakseimbangan data. Dalam hal personalisasi, cGAN dapat membantu menciptakan rekomendasi yang unik dengan memodelkan hubungan tersembunyi antar artikel berdasarkan preferensi pengguna. Dengan kemampuan ini, cGAN berkontribusi besar dalam menciptakan sistem klasifikasi yang lebih adaptif, akurat, dan relevan bagi pengguna (Wang, She, dan Ward 2021).

Metode kedua yang akan diimplementasikan dalam penelitian ini adalah *Bidirectional Encoder Representations from Transformers* (BERT). BERT adalah model berbasis *transformer* yang memungkinkan pemahaman konteks dalam dua arah, yakni dari kiri ke kanan dan sebaliknya(Rogers, Kovaleva, and Rumshisky 2020). Dalam penelitian ini, BERT berfungsi untuk mendukung proses klasifikasi artikel ilmiah dengan memahami konteks teks secara mendalam. Dengan pendekatan *bidirectional*, BERT dapat menganalisis data teks dari artikel ilmiah, dalam penelitian ini digunakan judul dan abstrak untuk menentukan kategori atau subjek penelitian secara lebih akurat (Adhikari dkk., 2020). Selain itu, BERT digunakan untuk meningkatkan efisiensi dan akurasi dalam pengolahan data dalam jumlah besar tanpa memerlukan pelabelan manual secara menyeluruh, menjadikannya alat penting dalam tugas klasifikasi berbasis teks di platform GARUDA.

Dengan mempertimbangkan keunggulan kedua metode tersebut, penelitian ini bertujuan untuk mengimplementasikan cGAN dan BERT dalam sistem klasifiksi publikasi jurnal terindeks GARUDA dengan harapan dapat membantu meningkatkan relevansi hasil pencarian serta memberikan prediksi yang lebih akurat berdasarkan preferensi dan kebutuhan peneliti.

## 1.2 Perumusan Masalah

Penelitian ini dilatar belakangi oleh beberapa permasalahan berikut:

1. Ketidakseimbangan jumlah publikasi antar bidang ilmu menyebabkan model klasifikasi cenderung bias terhadap kelas mayoritas.
2. Bagaimana menerapkan metode cGAN dan BERT untuk membangun sistem klasifikasi bidang ilmu artikel ilmiah pada platform Garuda secara tepat, dengan mempertimbangkan distribusi topik yang tidak seimbang serta keterbatasan informasi yang tersedia?

## 1.3 Pembatasan Masalah

Adapun penentuan masalah pada penelitian sebagai berikut :

1. Penelitian ini hanya menggunakan 5 bidang ilmu artikel ilmiah pada *platform* Garuda yaitu *Economics*, *Education, Humanities, Social Science*, dan *Language*.
2. Analisis data dilakukan hanya pada atribut tertentu yaitu judul dan abstrak.

## 1.4 Tujuan

Tujuan dari penelitian ini adalah untuk mengimplementasikan cGAN dan BERT sebagai metode dalam mengklasifikasikan jurnal artikel ilmiah yang tersedia pada *platform* Garuda.

## 1.5 Manfaat

Manfaat yang diharapkan dalam penelitian ini adalah tersusunnya sistem klasifikasi bidang ilmu artikel ilmiah menggunakan metode cGAN dan BERT yang mampu mengelompokkan artikel secara lebih terstruktur berdasarkan bidang ilmunya. Sistem ini diharapkan dapat mempermudah peneliti dalam menemukan jurnal yang sesuai dengan bidang kajian yang diteliti.

## 1.6 Sistematika Penulisan

Sistematika penulisan yang akan digunakan oleh penulis dalam sebuah laporan tugas akhir adalah sebagai berikut:

BAB I : PENDAHULUAN

Pada bab ini, penulis menjelaskan urgensi penelitian yang dilakukan, dimulai dengan penyusunan latar belakang, perumusan masalah, pembatasan masalah, serta penetapan tujuan dan manfaat yang diharapkan. Selain itu, sistematika penulisan juga disajikan untuk memberikan gambaran struktur penulisan secara keseluruhan.

BAB II : TINJAUAN PUSTAKA DAN DASAR TEORI

Bab ini memuat landasan teori yang mendukung penelitian serta referensi sebelumnya yang berkontribusi dalam perancangan sistem. Selain itu, bagian ini membantu penulis dalam memahami konsep yang berkaitan dengan *Conditional Generative Adversarial Network* (cGAN) dan *Bidirectional Encoder Representations from Transformers* (BERT)

BAB III : METODE PENELITIAN

Bab ini menjelaskan tahapan penelitian, dimualai dari proses pengumpulan data hingga tahap pemodelan yang digunakan dalam penelitian. Serta membahas analisis terhadap proses sistem yang akan dikembangkan, termasuk perancangan sistem *website* serta desain antarmuka (*Interface Design*) yang dirancang.

BAB IV : HASIL DAN ANALISIS PENELITIAN

Bab ini menyajikan temuan penelitian, yaitu hasil *Conditional Generative Adversarial Network* (cGAN) dan *Bidirectional Encoder Representations from Transformers* (BERT) sebagai metode dalam mengklasifikasikan jurnal artikel ilmiah yang tersedia pada *platform* Garuda.

BAB V : KESIMPULAN DAN SARAN

Bab ini, penulis menyajikan kesimpulan dan saran dari penulis terhadap penelitian yang telah dilakukan.

# BAB II
# TINJAUAN PUSTAKA DAN DASAR TEORI

## 2.1 Tinjauan Pustaka

Dalam penelitian yang dilakukan terdapat beberapa sumber literatur yang menjadi acuan penulis. Pertama, penelitian mengenai klasifikasi bidang fokus publikasi jurnal penelitian yang terindeks pada portal Garuda dengan metode *Multiclass Support Vector Machines* (SVM). Memiliki tujuan untuk membuat model klasifikasi bidang fokus penelitian berupa jurnal yang terindeks pada portal Garuda dengan menggunakan metode *Multiclass Support Vector Machines* (SVM). Modul klasifikasi ini merupakan mekanisme sistem yang nantinya digunakan dalam penentuan klasifikasi dengan ruang lingkup pembagian publikasi karya ilmiah berupa jurnal-jurnal penelitian berdasarkan kategori bidang focus penelitian (Sa'adah 2022).

Selanjutnya, ada penelitian implementasi *Latent Dirichlet Allocation* (LDA) dan *K-Nearest Neighbors* (KNN) pada sistem rekomendasi jurnal terindeks GARUDA. *Latent Dirichlet Allocation* (LDA) memiliki keunggulan dalam mengelompokkan data dalam jumlah besar dibandingkan beberapa metode pemodelan topik lainnya. Selain itu, LDA dapat diterapkan untuk mengidentifikasi topik dalam jurnal ilmiah, serta digunakan dalam proses klasifikasi dan pengelompokan data (Oktafiandi 2023). Tujuan penelitian ini adalah untuk mengimplementasikan metode LDA dan metode KNN dalam mencari jurnal yang relevan dengan abstrak dan judul artikel yang telah dipublikasikan di Garuda (Anisatuzzumara 2024).

Penelitian mengenai sistem pencarian trend judul tugas akhir mahasiswa Teknik informatika Universitas Islam Sultan Agung (Unissula) menggunakan metode *keyword extraction*. Fokus utama penelitian ini adalah mengambil studi kasus dengan mengambil data skripsi dari situs repositori Unissula untuk meringkas dokumen berbahasa Indonesia. Hasil dari penelitian tersebut menghasilkan aplikasi pencarian karya ilmiah yang dapat menghasilkan bobot setiap karya ilmiah dan dapat menampikan karya ilmiah sesuai kata kunci yang dicari serta memverifikasi

dokumen yang mengandung kata kunci tersebut. Dataset yang digunakan merupakan hasil *scrapping extensions web scrapper* pada website repositori Unissula yang digunakan sebagai bahan penelitian, dimana pada website tersebut terdapat kumpulan skrispi mahasiswa Teknik Informatika dari tahun 2018 – 2022 (Supardi 2023).

Penelitian selanjutnya adalah mengenai Klasifikasi artikel-artikel jurnal pustakaloka berdasarkan skema *Journal Information Topic Architecture* (JITA). JITA merupakan model taksonomi atau pengelompokan topik dalam literatur ilmiah. Penelitian ini menyoroti pentingnya klasifikasi dalam pengorganisasian pengetahuan, terutama dalam konteks jurnal Pustakaloka yang telah mempublikasikan lebih dari 170 artikel sejak 2009 dan terakreditasi Sinta 3. Klasifikasi menjadi alat penting untuk mengelompokkan informasi berdasarkan aturan tertentu, yang mempermudah pemahaman dan penelusuran. Salah satu skema klasifikasi yang relevan dalam bidang perpustakaan adalah JITA. Dengan mengorganisasi dokumen berdasarkan hierarki subjek dari umum ke spesifik, dilengkapi kode alfabet atau numerik untuk kemudahan. Penerapan skema JITA di jurnal seperti Pustakaloka memungkinkan analisis mendalam atas berbagai topik yang dibahas, sehingga mendukung pengembangan ilmu perpustakaan dan informasi (Syarifudin 2022).

Selanjutnya adalah penelitian mengenai Klasifikasi Teks Transduktif dengan menggabungkan GCN dan BERT. Penelitian ini memperkenalkan BertGCN, sebuah model untuk klasifikasi teks yang menggabungkan kekuatan *pretraining* berskala besar dan pembelajaran transduktif menggunakan *graph convolutional networks* (GCN). BertGCN membangun *graph heterogen* di mana simpul (*nodes*) mewakili kata atau dokumen, sementara representasi simpul diinisialisasi menggunakan representasi BERT yang telah dilatih sebelumnya. BertGCN tidak hanya menggabungkan representasi awal BERT dengan kekuatan GCN tetapi juga memungkinkan pelatihan bersama antara modul BERT dan GCN. Hasilnya, model ini mencapai performa terbaik dalam berbagai dataset klasifikasi teks dibandingkan pendekatan sebelumnya (Kuang dkk., 2021).

Penelitian mengenai *Conditional Generative Adversarial Networks*(cGANs) dan *Deep Learning Data Augmentation* (LDA) mengevaluasi efektivitas cGAN dalam augmentasi data dalam berbagai bidang, termasuk untuk medis dan penginderaan jauh. Hasil penelitian tersebut adalah peningkatan akurasi hingga 17% pada dataset kecil melalui augmentasi data, kemampuan generalisasi yang lebih baik dibandingkan metode augmentasi tradisional, dan juga fleksibelitasi dalam berbagai arsitektur jaringan seperti ResNet18 dan DenseNet121(Ribas dkk,. 2025).

## 2.2 Dasar Teori

### 2.2.1 Sistem Klasifikasi

Sistem klasifikasi merupakan sistem yang memproses pengelompokkan, mengumpulkan benda/entitas yang sama dan menyeleksi yang berbeda (Anggraeni dkk., 2021). Sistem klasifikasi berfungsi sebagai alat bantu dalam penyusunan dan pengelolaan suatu kelas agar dapat tersusun dengan sistematis,logis, dan terstruktur sehingga pengguna dapat lebih mudah menemukan informasi yang dibutuhkan (Anggraeni dkk., 2021). Klasifikasi adalah proses pemahaman mendalam mengenai objek yang melibatkan pembuatan urutan pada kelompok atau kelas, dan hal lain yang memberikan makna tertentu dari sebuah realitas (Syarifudin 2022).

Menurut (Syarifudin 2022) fungsi utama sistem klasifikasi dalam konteks ilmiah antara lain:

1. Mempermudah proses pencarian informasi berdasarkan topik atau subjek
2. Mengelola data dalam jumlah besar agar lebih terorganisir
3. Meningkatkan efisiensi dalam proses penemuan Kembali informasi
4. Memberikan struktur semantik yang lebih jelas antar dokumen.

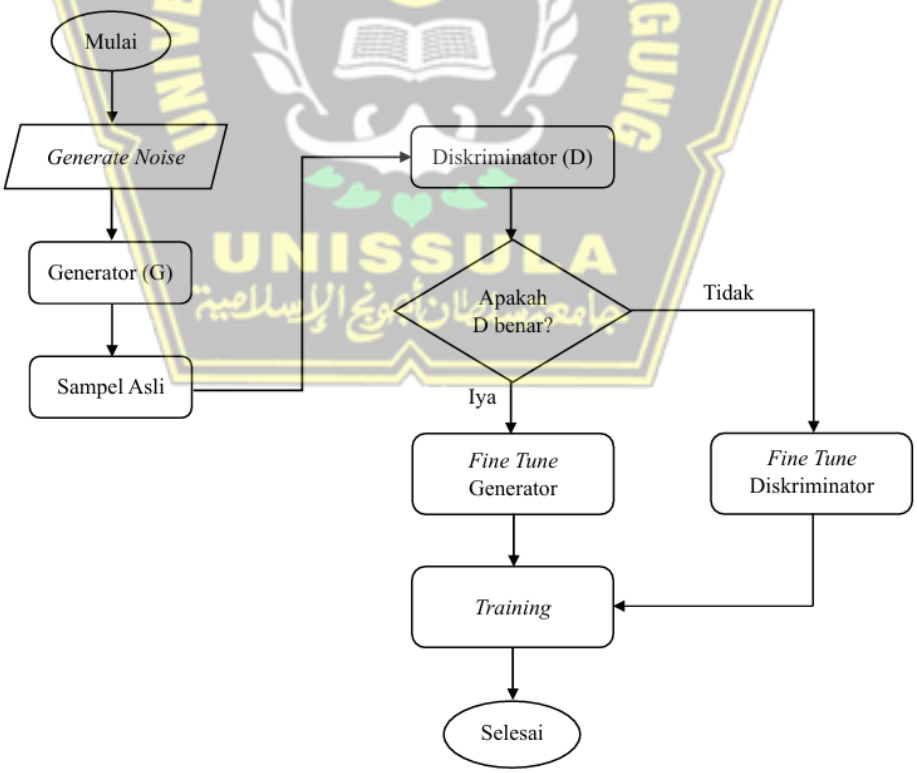
Integrasi kecerdasan buatan dalam sistem klasifikasi digital membantu mengatasi kendala skalabilitas dan meningkatkan efisiensi dalam mengelola ribuan data secara otomatis (Shah dkk., 2023). Oleh karena itu, dalam sistem klasifikasi menggunakan pendekatan NLP seperti GAN dan BERT digunakan untuk membangun sistem klasifikasi berbasis teks secara otomatis dan adaptif.

### 2.1.1 *Generative Adversarial Network* (GAN)

Kecerdasan buatan (AI) telah berkembang pesat, dengan *deep learning* menjadi metode utama untuk mengekstraksi fitur abstrak dari data. Teknologi ini

telah merevolusi pembelajaran tak terawasi, menawarkan solusi efektif untuk berbagai aplikasi tanpa memerlukan data berlabel, merangkum teori dasar, varian, teknik pelatihan, dan aplikasi GANs sambil mengulas tantangan dan peluang pengembangannya(Pan dkk., 2021).

GAN memiliki dua jaringan yang digunakan untuk membangun satu sama lain. Kumpulan jaringan pertama berperan sebagai pengklasifikasi, yang perlu dilatih agar dapat mengidentifikasi apakah yang dihasilkan asli atau tidak, sementara jaringan kedua berperan sebagai generator, yang menghasilkan sampel acak yang menyerupai sampel nyata dan menggunakannya sebagai sampel tiruan (Suprapti dkk., 2023). GAN menggunakan prinsip game teori sero-sum, Dimana kedua jaringan dilatih secara bersamaan dan saling memperbaiki. Seiring pelatihan, generator belajar menghasilkan data yang semakin mirip dengan data nyata, sementara discriminator semakin terlatih membedakan mana yang data asli dan mana data hasil buatan generator.




Gambar 2. 1 arsitektur GAN

Gambar 2.1 merupakan gambar arsitektur GAN. Berikut merupakan arsitektur GAN yang digunakan untuk penelitian ini:

1. *Generator*: GAN digunakan untuk menghasilkan artikel ilmiah sintetik yang realistis. Generator dilatih untuk menciptakan teks yang menyerupai artikel ilmiah asli dengan memanfaatkan dataset yang sudah ada.
2. *Discriminator*: Discriminator berperan untuk membedakan antara artikel ilmiah yang asli dan artikel ilmiah yang dibuat oleh generator. *Discriminator* ini akan memberikan umpan balik kepada generator untuk menghasilkan artikel sintetik yang lebih realistis.
3. Augmentasi Dataset: Artikel yang dihasilkan oleh GAN akan digunakan untuk memperluas dataset yang ada, sehingga model akan memiliki lebih banyak data *untuk* dilatih.

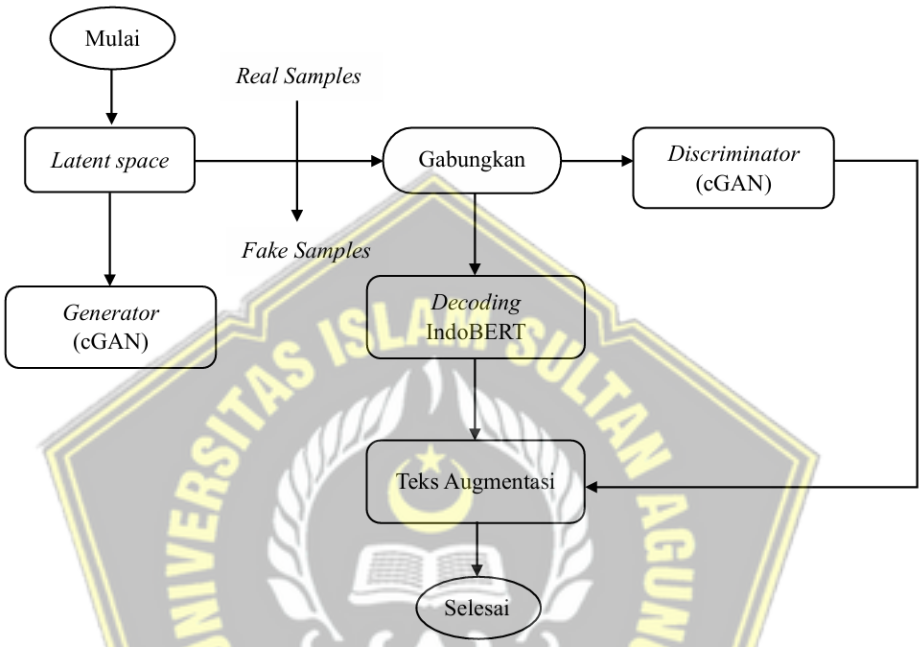
GAN efektif untuk menghasilkan data tambahan atau augmentasi ketika data pelatihan yang dimiliki masih terbatas dan kelasnya tidak seimbang (Comparison 2022). Berikut beberapa kelebihan yang dimiliki GAN yang dapat membantu proses penelitian:

1. Mengatasi *imbalanced* dataset dengan menghasilkan lebih banyak data dari kelas minoritas tanpa mengganggu distribusi data asli.
2. Tidak membutuhkan label manual sehingga cocok digunakan untuk memperluas dataset dengan biaya labelling rendah.
3. Artikel sintetik yang dihasilkan memiliki kemiripan semantik tinggi dengan artikel asli.

### 2.1.2 *Conditional Generative Adversarial Network* (cGAN)

Salah satu jenis model GAN akan digunakan dalam penelitian ini, yaitu *Conditional Generative Adversarial Network* (cGAN). cGAN merupakan pengembangan dari arsitektur GAN yang mengintrogasikan informasi tambahan berupa label atau kondisi tertentu ke dalam proses pembentukan kata. Dengan pendekatan ini, generator mampu meciptakan data yang selaras dengan kondisi yang diberikan. Dalam arsitektur cGAN baik *generator* maupun *discriminator* menerima input tambahan berupa kondisi yang memungkinkan control lebih spesifik terhadap data yang dihasilkan (Ma dan Qu 2022).

cGAN memiliki kemampuan generalisasi yang lebih baik dibandingkan dengan metode augmentasi tradisional. Arsitektur cGAN yang ditingkatkan dapat menghasilkan data tabular sintesis berkualitas tinggi dengan kemampuan menangani data campuran (numerik dan kategorikal) dengan lebih efektif yang mendekati distribusi data asli(Alqulaity dan Yang 2024).




Gambar 2. 2 arsitektur cGAN

Gambar 2. 2 merupakan arsitektur cGAN. Berikut merupakan representasi dari arsitektur cGAN dengan tujuan untuk mempelajari distribusi data nyata dari proses augmentasi.

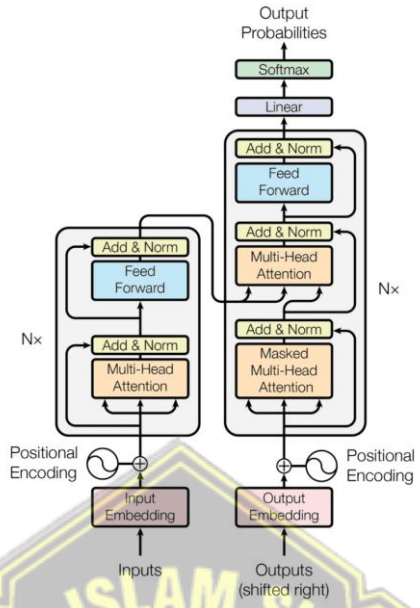
1. Langkah awal untuk proses augmentasi adalah *latent Space,* yang merupakan ruang vektor berdimensi tinggi yang berisi nilai acak yang menjadi input awal untuk *generator*. Nilai ini diambil dari distribusi acak.
2. Kemudian melewati proses *noise vector* dimana vektor acak diambil untuk membuat sampel data palsu.
3. Masuk kedalam proses *generator* dimana model *neural network* bertugas untuk menghasilkan data sintetik dengan mengubah noise menjadi data palsu yang menyerupai data asli.
4. *Real samples* atau data dari dataset yang sebenarnya digunakan sebagai pembanding terhadap output dari *generator*.

5. Selanjutnya *real samples* dan *generated samples* dikombinasikan tanpa memberi tahu apakah data tersebut data asli atau palsu, fungsinya adalah agar *discriminator* dapat belajar membedakan data tanpa label ekspilisit.
6. *Discriminator* sebagai model *neural network* kedua yang bertugas menilai apakah input adalah data asli atau hasil buatan generator.
7. Proses *is* D *correct*? adalah untuk menilai loss dari *discriminator*.
8. Antara *discriminator* dan *generator* terus meningkatkan kemampuan hingga menghasilkan data palsu dengan nilai akurasi tinggi meyerupai data asli.

### 2.1.3 *Transformers*

*Transformers* adalah arsitekur jaringan saraf yang diciptakakan untuk memproses data sekuensial dengan mengandalkan mekanisme *self-attention*, dengan itu memungkinkan *transformers* untuk memproses secara paralel sehingga lebih efisien dalam menangani dependensi jangka panjang dalam data (Islam dkk., 2023). Mekanisme *self-attention* memungkinkan model untuk mempertimbangkan hubungan antara semua token dalam input, tanpa memandang jarak dan posisi. *Self-attention* menghitung nilai perhatian antara setiap pasangan token, menungkinkan model untuk menangkap dependensi secara efisien (Chandra dkk., 2023).

*Transformers* sendiri didefinisikan sebagai model yang terdiri dari *encoder* dan *decoder* berbasis *attention mechanism*, yang dapat mengatasi keterbatasan konteks pendek pada model-model sebelumnya (Islam dkk., 2023). *Encoder* berfungsi untuk memahami dan mepresentasikan input teks ke dalam bentuk vektor (*embedding*) yang mengandung informasi semantik. Sedangkan *decoder* digunakan untuk menghasilkan output teks seperti pada *text generation*. *Transformers* unggul dalam tugas-tugas NLP seperti klasifikasi teks, terjemahan, dan penjawaban pertanyaan.




Gambar 2. 3 Arsitektur *Transformers* (Vaswani dkk., 2017)

Gambar 2. 2 merupakan gambar arsitektur dari *transformers*. Setiap layer dalam *encoder* dan *decoder* terdiri dari sub layer sebagai berikut:

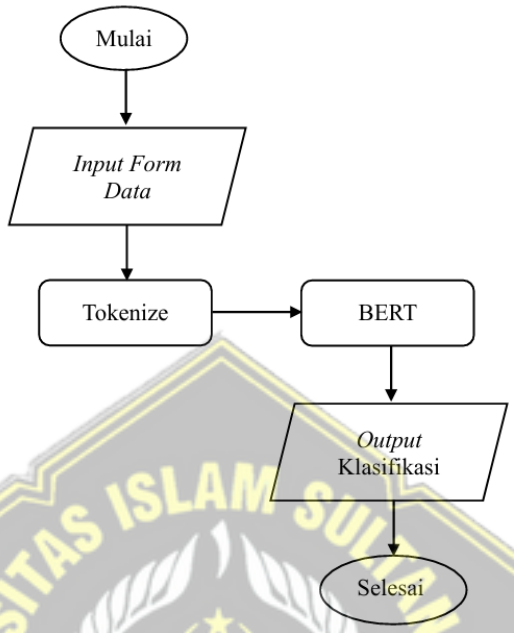
1. *Multi-Head Self-Attention*, memungkinkan model untuk focus pada bagian-bagian berbeda dari input.
2. *Feed-Forward Neural Network* (FFN), yaitu lapisan jaringan saraf yang diterapkan secara identic pada setiap posisi.
3. *Add and Norm*, merupakan mekanisme normalisasi layer.

Menurut (Islam dkk., 2023), transformers telah menjadi arsitektur dominan dam NLP karena kemampuannya yang superior dalam memahami konteks dan struktur bahasa.

### 2.1.4 *Bidirectional Encoder Representasions From Transformers (BERT)*

Salah satu algoritma *deep learning* yang dikembangkan oleh *Google* adalah *Bidirectional Encoder Representations from Transformers* atau BERT. BERT adalah model representasi hahasa berbasis *transformers* yang dilatih untuk memahami konteks secara simultan (Kenton dkk,. 2022). Algoritma BERT memproses konteks penuh dengan melihat pola yang muncul sebelum atau sesudah kata untuk memproses kata (Nayla dkk., 2023). BERT memahami teks dengan cara *bidirectional*, yaitu membaca konteks dari kiri ke kanan dan kanan ke kiri secara

bersamaan. Hal ini memungkinkan BERT memahami makna kata secara lebih kontekstual, terutama dalam kalimat konteks.




Gambar 2. 4 arsitektur BERT

Arsitektur BERT terdiri dari beberapa lapisan *encoder transformers*, model ini tersedia dengan dua ukuran utama, yaitu:

1. BERT_BASE: 12 lapisan (*layers*), 768 dimensi hidden, dan 12 kepala perhatian (*attention heads*), dengan total 110 juta parameter.
2. BERT_LARGE: 24 lapisan, 1024 dimensi hidden, 16 kepala perhatian, dengan total 340 juta parameter.

*Encoding* dengan BERT melewati beberapa proses, yaitu:

1. *Layer Transformer*: Teks melalui beberapa lapisan transformer (biasanya 12 atau 24 lapisan tergantung pada versi BERT). Di setiap lapisan, model memahami hubungan antar kata berdasarkan konteks dua arah (bidirectional). Misalnya, BERT memahami bahwa kata "bank" dalam "bank of the river" memiliki arti yang berbeda dibandingkan "bank account".
2. *Self-Attention Mechanism*: Setiap token berinteraksi dengan token lain dalam teks menggunakan mekanisme perhatian (*attention*). Ini memungkinkan model memahami konteks global *teks*.

BERT dilatih melalui dua tahap utama yaitu *pre-training* dan *fine-tuning*.

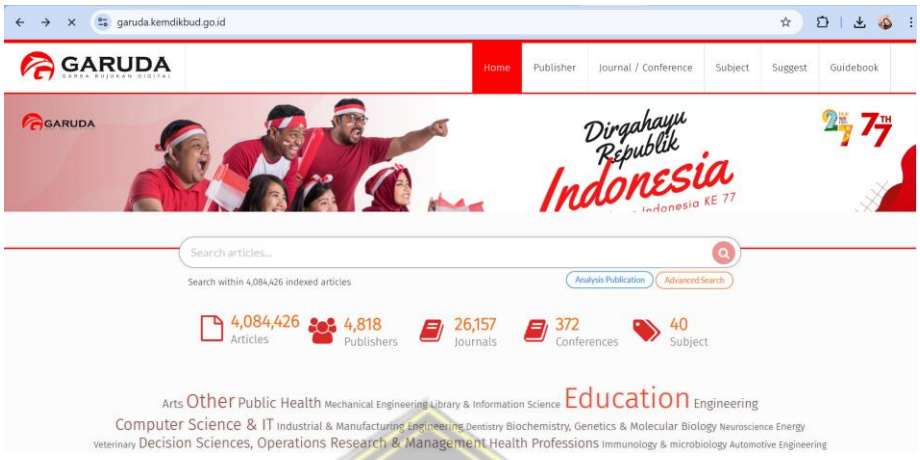
1. *Pre-training*: Pada tahap ini, BERT dilatih pada data teks besar tanpa label menggunakan dua tugas, yaitu *Masked Language Modeling* (MLM) dengan fungsi sebagian token dalam input digantikan dengan token khusus [MASK], dan model belajar untuk memprediksi token asli berdasarkan konteks sekitarnya. Kemudian *Next Sentence Prediction* (NSP) sebagai model belajar untuk memprediksi apakah dua kalimat berurutan dalam data benar-benar berurutan atau tidak, membantu dalam pemahaman hubungan antar kalimat.
2. *Fine-tuning*: Setelah *pre-training*, BERT dapat disesuaikan untuk tugas spesifik seperti klasifikasi teks, penjawaban pertanyaan, dan analisis sentimen dengan menambahkan lapisan output tambahan dan melatih model pada data berlabel.

### 2.1.5 IndoBERT

Salah satu jenis *transformers* yang terkenal dan digunakan dalam penelitian ini adalah indoBERT. IndoBERT adalah sebuah *pretrained language model* berbasis arsitektur BERT yang diadaptasi khusus untuk bahasa Indonesia. Model ini dikembangkan oleh IndoNLU dan pertama kali diperkenalkan oleh tim IndoBERT Benchmark (IndoBenchmark). IndoBERT merupakan model pra-latih yang fokus menghasilkan data augmentasi berbahasa Indonesia dengan kualitas tinggi. Keunggulannya adalah tidak perlu training dari nol karena IndoBERT berbasis BertModel bisa langsung menggunakan API *transformers* untuk tokenisasi, *feature extraction*, dan preprocessing cepat.

Model IndoBERT telah melalu pelatihan dengan lebih dari 220 juta kata berbahasa indonesia yang diambil dari berbagai sumber seperti Wikipedia (70 juta), artikel-artikel Indonesia (50 juta), dan Corpus Indonesia (90 juta) (Widiansyah dkk., 2021). Untuk meberikan kinerja model terbaik akan dilakukan pengujian skenario dengan menambahkan dan mengatur parameter-parameter untuk pemrosesan data. IndoBERT tersedia dalam beberapa versi, salah satu jenis yang digunakan untuk penelitian ini adalah model indobenchmark atau indobert-base-p1, yang dikembangkan dan dipublikasikan di *platform HuggingFace*.

### 2.1.6 Garba Rujukan Digital GARUDA




Gambar 2. 5 platform Garuda

Garuda (Garba Rujukan Digital) adalah sistem basis data referensi digital yang dikembangkan oleh Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi Indonesia. Tujuan utamanya adalah untuk mempermudah akses terhadap dokumen akademik seperti skripsi, tesis, disertasi, dan artikel ilmiah lainnya. Garuda memfasilitasi penyebaran karya akademik, mendukung gerakan akses terbuka, dan mempermudah manajemen serta distribusi artikel penelitian.

Garuda menyediakan akses penuh ke berbagai dokumen penelitian yang terindeks, memudahkan peneliti dan masyarakat untuk mencari dan mengutip karya ilmiah (Wijaya dan Negara 2022). Sistem ini sangat penting dalam konteks pendidikan tinggi di Indonesia, terutama untuk mempermudah pengelolaan dan penyebaran informasi akademik di perguruan tinggi dan lembaga penelitian. Dengan Garuda, perguruan tinggi dapat meningkatkan infrastruktur akademiknya, baik dalam pengelolaan koleksi maupun dalam mendistribusikan artikel ilmiah secara digital.

Tujuan adanya platform ini yaitu untuk memfasilitasi ketersediaan informasi umum dan ilmiah yang dapat diakses dengan mudah dan cepat, membangun jaringan perpustakaan digital di Indonesia. Adanya portal sumber informasi tersebut, menjadi sumber referensi yang lebih luas akan tersedia, yang tidak hanya berasal dari buku tetapi juga dari artikel jurnal yang paling baru yang mengikuti perkembangan ilmu terbaru(Anisatuzzumara, 2024)

# BAB III
# METODE PENELITIAN

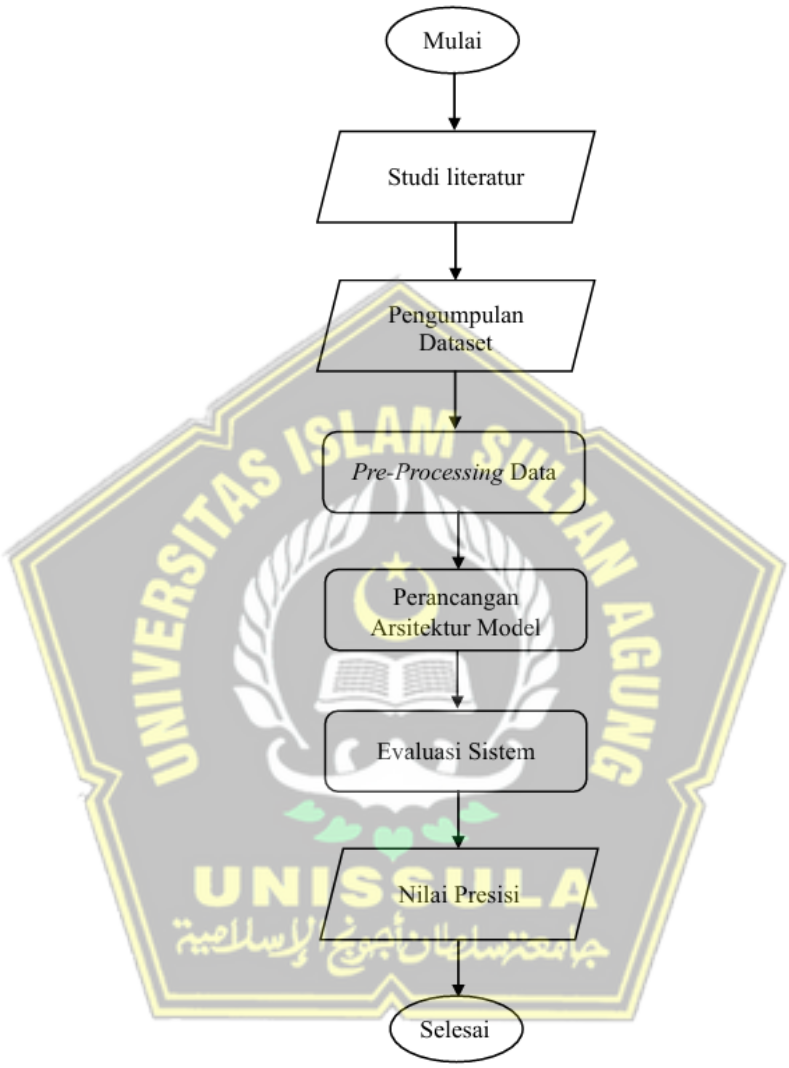
## 3.1 Metode Penelitian

Penelitian ini menggunakan metode klasifikasi berbasis *machine learning Natural Language Processing* (*NLP*) yaitu *Conditional Generative Adversarial* Network (cGAN) dan *Bidirectional Encoder Representations from Transformers* (BERT) digabungkan untuk menghasilkan model klasifikasi yang lebih akurat terhadap artikel ilmiah yang terindeks Garuda.

cGAN digunakan untuk menangani ketidakseimbangan data dalam dataset dengan menghasilkan data sintetik yang menyerupai data asli dari kategori minoritas, sehingga memperbesar dan memperbaiki distribusi dataset. BERT digunakan untuk melatih model klasifikasi, di mana representasi teks yang kaya dari BERT akan membantu memahami konteks artikel dengan lebih baik. Integrasi keduanya dilakukan dengan memanfaatkan data hasil augmentasi dari cGAN sebagai input tambahan dalam pelatihan BERT, sehingga data yang lebih beragam dapat meningkatkan kemampuan generalisasi dan akurasi klasifikasi model secara keseluruhan.

Penggabungan ini memberikan keuntungan utama, yaitu memanfaatkan data tak berlabel dan data sintetik untuk meningkatkan performa BERT dalam klasifikasi dokumen, bahkan dengan dataset berlabel yang terbatas (Croce dkk., 2020).

Berikut merupakan gambaran alur *flowchart* untuk proses penelitian ini dari pengumpulan data hingga diperoleh hasil untuk implementasi metode cGAN - BERT:




Gambar 2. 6 Alur Penelitian

Gambar 3. 1 merupakan alur dari penelitian klasifikasi artikel ilmiah terindeks Garuda dengan metode cGAN – BERT dengan penjelasan lebih lanjut sebagai berikut:

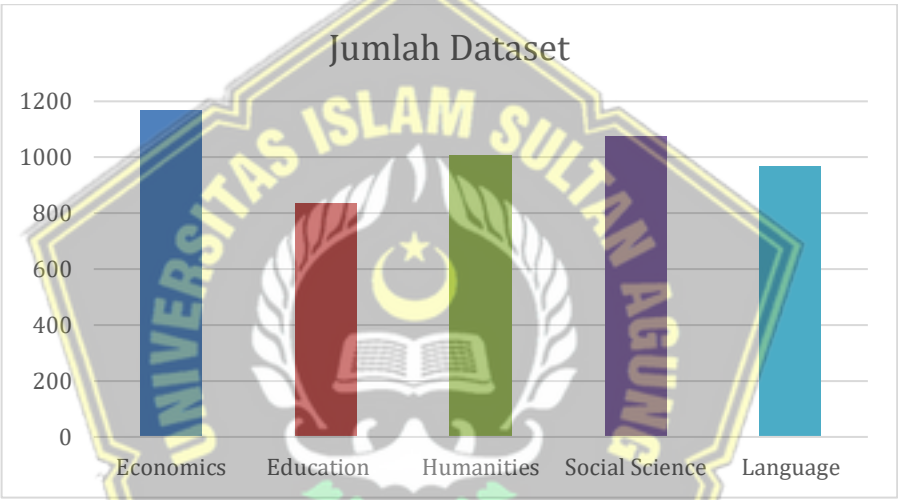
### 3.1.1 Studi Literatur

Dengan melakukan studi literasi, penulis dapat mempelajari teori *text mining* tentang *Preprocessing Text* baik berupa proses pada *Natural Language Processing* (NLP) serta implementasi metode cGAN dan BERT dari berbagai sumber literasi

seperti tesis, jurnal, makalah, skripsi, ataupun situs-situs *website* yang akan diulas untuk mempelajari teori tersebut.

### 3.1.2 Pengumpulan Dataset

Pada tahap pengumpulan data ini bersumber dari dataset pada *platform* Garuda. Teknik pengumpulan data pada penelitian ini dilakukan dengan metode *web scrapping*, yaitu teknik ekstrasi data dari halaman *web* secara otomatis menggunakan bantuan bahasa pemrograman. Proses ini mengidentifikasi struktur web terutama pada bagian yang menampilkan daftar artikel, judul, dan abstrak yang kemudian akan diambil sebagai bahan dataset penelitian.

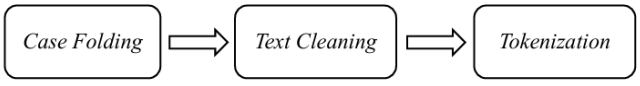



Gambar 2. 7 *chart* dataset hasil *scraping*

Gambar 3. 2 merupakan *chart* dataset setelah *scraping* data. Dengan mengambil judul dan abstrak publikasi dari artikel ilmiah total data hasil *scraping* adalah 4766 artikel dari 23 jurnal terindeks garuda dengan jumlah per-datanya yaitu *Economics* 1166 data, *Education* 834 data, *Humanities* 1008 data, *Social Science* 1075 data , dan *Language* 968 data.

### 3.1.3 *Pre-Processing* Data

Pada tahap ini, peneliti merencanakan proses perancangan model. Berikut ini merupakan diagram tahapan dalam memproses teks dimana tahapan ini akan digunakan untuk proses penelitian.




Gambar 2. 8 Tahapan *pre-processing* data

Adapun penjelasan rinci mengenai tahapan – tahapan di atas:

1. *Case Folding*

   Pada tahap ini, setiap kata dalam teks akan diubah menjadi huruf kecil menggunakan fungsi seperti *lower*() dalam *Python*. Ini mengurangi ketidakcocokan akibat perbedaan kapitalisasi antara kata yang serupa terutama huruf atau karakter alphabet menjadi huruf kecil.

   Tabel 3. 1 contoh *case folding*

| Data sebelum | Data sesudah |
|---|---|
| COVID-19 merupakan penyakit yang disebabkan oleh infeksi dari Virus Severe Acute Respiraotry Syndrome Coronavirus 2 (SARS-COV-2) | covid-19 merupakan penyakit yang disebabkan oleh infeksi dari virus severe acute respiraotry syndrome coronavirus 2 (sars-cov-2) |

2. *Text Cleaning*

   Tahapan *text cleaning* adalah tahap untuk penghapusan tanda baca yang tidak relevan, penghapusan angka yang tidak memberikan informasi penting untuk klasifikasi, dan spasi berlebih untuk meningkatkan kualitas teks.

   Tabel 3. 2 contoh text cleaning

| Data sebelum | Data sesudah |
|---|---|
| covid-19 merupakan penyakit yang disebabkan oleh infeksi dari virus severe acute respiraotry syndrome coronavirus 2 (sars-cov-2) | covid 19 merupakan penyakit yang disebabkan oleh infeksi dari virus severe acute respiraotry syndrome coronavirus 2 sars cov 2 |

3. *Tokenization*

   Tokenisasi merupakan proses pemecahan teks menjadi unit-unit kecil yang disebut token, yang bisa berupa kata, frasa, atau bahkan karakter. Tokenisasi memudahkan model untuk memproses teks, karena model bekerja dengan token, bukan teks mentah.

Tabel 3. 3 contoh tokenisasi

| Data sebelum | Data sesudah |
|---|---|
| covid-19 merupakan penyakit yang disebabkan oleh infeksi dari virus severe acute respiraotry syndrome coronavirus 2 (sars-cov-2) | [“covid-19”, “merupakan”, “penyakit”, “yang”, “disebabkan”, “oleh”, “infeksi”, “dari”, “virus”, “severe”, “acute”, “respiraotry”, “syndrome”, “coronavirus”, “2”, “(sars-cov-2)” |

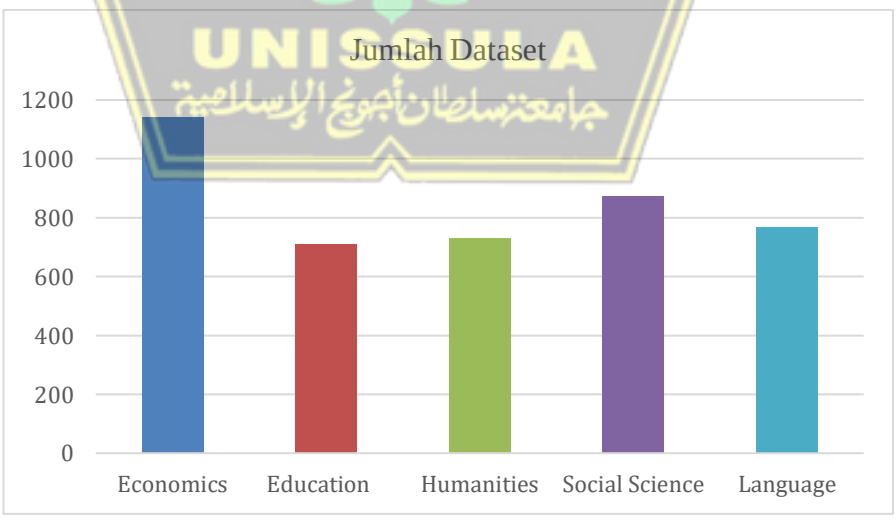
Proses pembersihan data hanya dilakukan dengan langkah - langkah di atas dikarenakan memiliki beberapa alasan, antara lain:

1. Menjaga keaslian data

   Untuk digunakan pada sistem klasifikasi, penting untuk menjaga keaslian data sehingga model mampu menghasilkan variasi data yang relevan dan semantic dengan mempertahankan konteks aslinya.

2. Menghindari perubahan yang tak terduga pada makna teks

   Jika proses pembersihan dilakukan juga menggunakan *stemming* atau pergantian sinonim dan juga *stopwords* atau penghapusan kata tidak deskriptif, hal tersebut akan berpotensi mengubah makna/ konteks dari teks asli sehingga akan membuat proses embedding teks menjadi sulit dilakukan.




Gambar 2. 9 *chart* dataset *pre-processing*

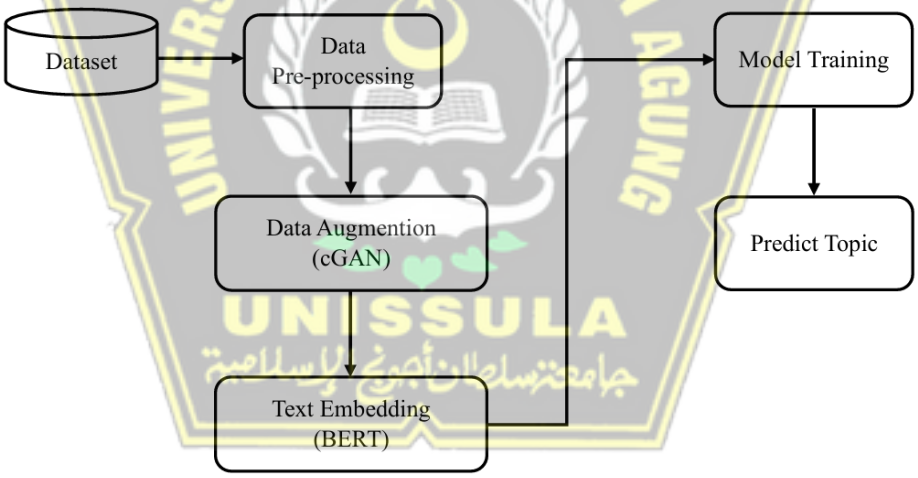
Gambar 3. 4 merupakan jumlah dataset setelah *pre-processing* data. Terdapat penurunan jumlah sebanyak 540 data sehingga jumlah data menjadi 4226 data

dengan rincian *Economics* 1142 data, *Education* 711 data, *Humanities* 730 data, *Social Science* 874 data, dan *Language* 769 data. Diperoleh hasil bahwa bidang ilmu *Economics* memiliki jumlah dataset terbanyak sehingga jumlah tersebut akan digunakan sebagai acuan jumlah data augmentasi yang akan dibuat agar ke lima bidang tersebut seimbang.

### 3.1.4 Perancangan Arsitektur Model

Dalam tahap ini, akan direncanakan langkah-langkah proses perancangan arsitektur model secara rinci untuk memastikan setiap komponen dapat saling terintegrasi dengan baik. Setiap langkah dirancang dengan mempertimbangkan tujuan utama, yaitu menghasilkan model yang akurat, efisien, dan dapat diandalkan untuk merekomendasikan publikasi jurnal sesuai dengan kebutuhan pengguna.

Berikut merupakan gambar alur perancangan arsitektur model agar lebih mudah dipahami:




Gambar 2. 10 alur perancangan arsitekur model

1. *Data Augmentation* (cGAN)

Pada tahap ini, data dari hasil *pre-processing* akan diolah cGAN untuk menghasilkan data sintetik guna memperkaya dataset yang ada sehingga hasil akhirnya tiap label memiliki jumlah dataset yang seimbang. cGAN dipilih karena pada penelitian ini hanya menggunakan 5 label bidang ilmu sehingga mengharuskan terjadinya sebuah proses pengkondisionalan.

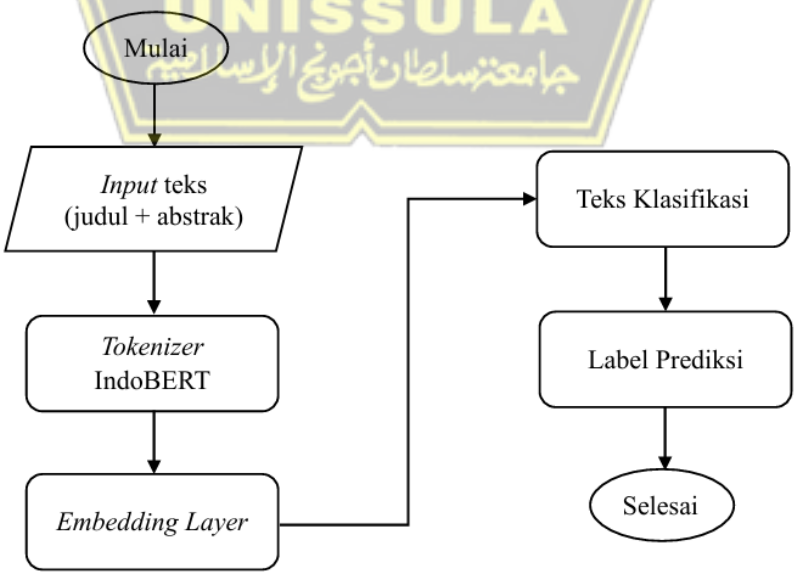
cGAN memiliki beberapa keunggulan dibandingkan dengan GAN konvensional, yaitu:

1. Kemampuan untuk mengondisikan *output*, memungkinkan kontrol lebih spesifik terhadap data yang dihasilkan yaitu berupa teks dengan topik yang relevan.
2. cGAN efektif digunakan untuk augmentasi data, terutama dalam kasus dimana data asli terbatas.
3. Dengan mengondisikan pada informasi tambahan, cGAN mampu menghasilkan data sintetik yang lebih realistis dan sesuai dengan distribusi asli.

Dalam penelitian ini, augmentasi cGAN tidak dilakukan pada bidang ilmu dengan jumlah data tertinggi yaitu *Economics* dengan jumlah 1142. Hal ini dikarenakan jumlah data tersebut sudah cukup efektif untuk digunakan dalam proses *training* BERT, sehingga proses augmentasi data sintetik hanya akan *diimplementasikan* pada 4 bidang ilmu lainnya yaitu *Education*, *Humanities*, *Social Science*, dan *Language*.

2. *Text Embedding* (BERT)

Proses Text Embedding dengan BERT melibatkan langkah-langkah yang memungkinkan model menghasilkan representasi semantik dalam bentuk vektor numerik (*embedding*) dari teks input. Representasi ini menangkap makna kata dan hubungan antar kata dalam konteks kalimat.



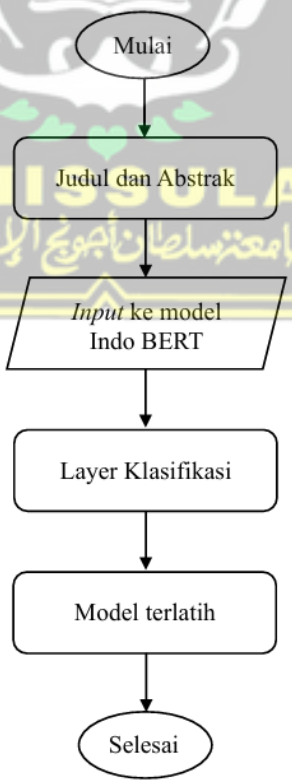

Gambar 2. 11 klasifikasi kalimat dengan BERT

Gambar 3. 7 merupakan representasi klasifikasi kalimat menggunakan BERT, dengan penjelasan yang lebih detail sebagai berikut:

1. Dengan menggunakan dataset yang telah di augmentasi, teks akan dipecah menjadi token-token oleh tokenizer BERT yang kemudian dikonversi menjadi angka (id) yang dapat diproses oleh BERT.
2. Kemudian id tersebut diubah menjadi representasi vektor berdimensi tinggi yang memuat informasi kata, posisi dalam kalimat, dan tipe segmen kalimat.
3. Semua token *embedding* masuk ke BERT encoder untuk diproses secara paralel dengan perhatian penuh ke semua token lainnya (*self-attention*).

3. Model *Training* (*Clasissier*) dengan IndoBERT

Proses pelatihan model klasifikasi menggunakan IndoBERT dilakukan dengan memanfaatkan *pre-trained* model indobenchmark/indobert-base-p1 yang telah dilatih sebelumnya menggunakan korpus bahasa Indonesia. Model ini kemudian di-*fine-tune* untuk tugas klasifikasi topik jurnal ilmiah berdasarkan input teks berupa judul dan abstrak artikel.




Gambar 2. 12 alur model training

Gambar diatas merupakan tahapan proses model latih dengan indoBERT, meliputi;

1. *Input* judul dan abstrak, dimana sistem memasukkan kombinasi judul dan abstrak artikel ilmiah sebagai teks masukan yang akan dikonversi menjadi unit-unit tokenizer dan diproses model menjadi id token numerik.
2. *Input* token ke model indoBERT, token id yang dihasilkan kemudian dimasukkan ke dalam model indoBERT. Model ini akan membaca input secara bidirectional dan menghasikan representasi vector dari seluruh kalimat.
3. *Layer* klasifikasi (*Dense* dan *softmax*), *output* dari token diteruskan ke *layer dense* untuk menghasilkan distribusi probabilitas ke dalam 5 bidang ilmu.
4. Setelah proses pelatihan selesai, model IndoBERT telah di-finetune khusus untuk tugas klasifikasi bidang ilmu artikel ilmiah berbasis data Garuda.

### 3.1.5 Evaluasi Model

Setelah sistem selesai dibangun, dilakukan penilaian menyeluruh untuk menjamin bahwa sistem berfungsi dengan baik dan mencapai sasaran penelitian. Evaluasi dimulai dengan menguji kinerja model dalam sistem menggunakan masukan dari pengguna. Langkah ini ditujukan untuk mengevaluasi apakah Solusi yang dihasilkan relevan dan sesuai dalam mengatasi masalah yang ada, tidak hanya berjalan secara teknis, tetapi juga mampu menhasilkan prediksi yang relevan, akurat, dan sesuai dengan tujuan penelitian.

1. Pengujian Model:

Tahap awal dari evaluasi dimulai dengan melakukan pengujian terhadap model BERT yang telah dilatih. Terdapat beberapa tahapan untuk proses tersebut yaitu:

a. Tokenisasi: Proses tokenisasi jurnal artikel menggunakan tokenizer BERT.
b. *Embedding*: setelah tokenisasi maka token yang dihasilkan akan diubah menjadi *vector embedding*.
c. Prediksi: embedding diklasifikasikan kedalam salah satu topik yang telah ditentukan yaitu *Economics, Education, Humanities, Social Science*, dan *Language*.

Dari ke tiga proses diatas, maka data sudah siap untuk dilanjutkan ke tahap evaluasi hasil klasifikasi bidang ilmu untuk mengetahui metrik evaluasinya.

2. Evaluasi Hasil Klasifikasi:

Setelah model selesai dilatih, metrik evaluasi diperlukan untuk mengetahui seberapa akurat model memprediksi topik artikel ilmihah. Metrik ini berdasarkan pada perbandingan antara hasil prediksi model dengan data yang sebenarnya.

a. Akurasi (Accuracy)

Akurasi melihat berapa banyak total prediksi model yang benar dibandingkan dengan semua percobaan

$$Accuracy = \frac{Jumlah\ Prediksi\ benar}{Total\ seluruh\ data} \quad (1)$$

Di mana:

1. TP = True Positive (prediksi benar dan aktual benar)
2. TN = True Negative (prediksi salah dan aktual salah)
3. FP = False Positive (prediksi salah tapi aktual salah satu topik lain)
4. FN = False Negative (gagal memprediksi topik yang benar)

b. Presisi (*Precision*)

Presisi digunakan untuk mengukur seberapa akurat model saat memprediksi suatu kelas tertentu.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

c. *Recall* (*Sensitivity*)

Mengukur seberapa baik model menangkap semua contoh dari suatu kelas.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

d. F1-*Score*

Merupakan harmonisasi antara presisi dan recall.

$$\text{F1-}score = 2\ x\ \frac{Precision\ x\ Recall}{Precision + Recall} \quad (4)$$

e. *Confusion Matrix*

Merupakan matriks yang menunjukkan performa model dalam membedakan antar kelas.

## 3.2 Analisis Sistem

Pada analisis sistem penelitian ini, penulis akan membuat sistem klasifikasi bidang ilmu artikel ilmiah terindeks Garuda. Untuk merancang alur sistem, diperlukan *flowchart* yang menunjukkan Langkah-langkah bagaimana sistem berjalan. Pada gambar 3.9 merupakan *flowchart* rancangan alur sistem yang akan dibangun :


Gambar 2. 13 Perancangan sistem

Gambar 3. 9 merupakan *flowchart* tahapan alur kerja sistem dilakukan dengan tahapan dibawah:

1. Pengguna terlebih dahulu mengakses halaman utama antar muka untuk menampilkan proses kerja dari aplikasi sistem presiksi solusi.
2. Terdapat tampilan sistem dengan untuk klasifikasi bidang ilmu artikel terindeks Garuda.

3. Pengguna atau sistem memasukkan kombinasi judul dan abstrak artikel ilmiah sebagai teks input yang akan diteliti.
4. Setelah pengguna memasukkan data , sistem akan menggabungkan data menjadi satu string teks untuk dianalisis model menggunakan indoBERT tokenizer yang kemudian akan menghasilkan output logits dari model menjadi probabilitas untuk setiap label.
5. Setelah proses prediksi selesai, sistem akan menampilkan hasil berupa tabel ranking probabilitas, dengan nilai probabilitas tertinggi sebagai prediksi topik utama yang paling relevan dengan data yang diteliti.
6. Setelah selesai ditampilkan, proses selesai. Pengguna dapat kembali ke tahap awal untuk memasukkan data yang lainnya.

### 3.3 Identifikasi Perangkat Lunak

Pada tahap analisis kebutuhan, peneliti menganalisis berbagai perangkat lunak yang diperlukan untuk mengembangkan sistem ini sehingga proses *input* hingga mengeluarkan hasil *output* sesuai dengan yang diharapkan. Berikut merupakan perangkat lunak yang digunakan untuk kebutuhan penelitian:

1. *Python* 3.13.2

   *Python* adalah jenis bahasa pemrograman komputer yang dapat digunakan untuk membuat situs web, *software* atau aplikasi dan analisis data yang bersifat *open source*. *Python* menggunakan sintaks yang mudah dipahami dan cenderung sederhana, sehingga mempercepat proses pengembangan serta mempermudah pemeliharaan kode (Cutting dan Stephen 2021). Bahasa *python* versi 3.13.2 yang akan digunakan untuk membuat sistem ini, yang merupakan versi minor dari python versi 3.13 yang dirilis oleh *Python Software Foundation* (PSF). Versi ini fokus pada perbaikan bug, peningkatan performa, dan beberapa penyesuaian internal tanpa menambahkan fitur besar baru (Dhruv dkk,. 2021).

2. *Google Colaboratory*

   *Google Colaboratory* (Colab) merupakan layanan *cloud* gratis dari *Google* yang memungkinkan *user* menulis dan menjalankan kode *python* melalui *web browser*. Dilengkapi dengan akses gratis terhadap *Graphics Processing Unit*

(GPU) dan *Tensor Processing Unit* (TPU) untuk proses yang lebih cepat dan intensif, colab sangat bermanfaat untuk proyek *machine learning* dan *deep learning* tanpa mengatur perangkat keras ataupun perangkat lunak di komputer lokal (Carneiro dkk., 2020). Colab terintegrasi dengan baik bersama *Google Drive* sehingga memudahkan untuk menyimpan dan membagikan *notebook* tempat kode *python* yang dibuat (Sharma dkk., 2021).

3. Pandas

   Pandas adalah *library* fundamental di *python* yang menawarkan struktur data dan alat analisis data yang tangguh dan mudah dipakai. Pandas digunakan untuk membaca berbagai format data seperti *Comma Separated Values* (CSV) dan Excel, membersihkan data yang hilang atau tidak konsisten, menyeleksi, mengurutkan, menggabungkan dataset, serta melakukan agregasi dan transformasi data untuk persiapan analisis atau *machine learning* (Snehkunj dkk., 2022).

4. *Visual Studio Code*

   Dalam kerangka sistem penelitian ini, *Visual Studio Code* (VSC) digunakan sebagai teks editor. Kemampuannya yang adaptif terhadap spektrum luas bahasa pemrograman dan f*ramework*, lintas *platform*, dan memiliki performa yang sangat cepat (Rask dkk., 2021). VSC memiliki fitur-fitur produktivitas seperti *IntelliSense* untuk *code completion* yang prediktif, alat *debugging* terintegrasi, dan konektivitas mulus dengan sistem kontrol versi Git. Dengan antarmuka yang efisien dan kemampuan adaptasi tinggi, VSC menjadi instrument vital untuk pembuatan sistem perangkat lunak dalam penelitian ini.

5. NumPy

   *Numerical Python* (NumPy) digunakan sebagai *library* inti untuk komputasi ilmiah pada *python* yang menyediakan dukungan untuk objek *array* multidimensional yang besar dan efisien. NumPy memungkinkan operasi matematika yang sangat cepat pada *array* data seperti penjumlahan, perkalian matriks, dan trigonometri yang jauh lebih efisien(Van Der Walt dkk., 2021). NumPy digunakan dalam tahap *pre-processing* data, seperti konversi data hasil

ekstrasi fitur kedalam format *array* numerik untuk mempercepat proses perhitungan *embedding* atau menghasilkan data sintetis dari model cGAN.

6. PyTorch

   PyTorch merupakan *library* utama dalam membangun dan melatih model cGAN serta model klasifikasi IndoBERT. PyTorch digunakan untuk mendefinisikan arsitektur *neural network* atau jaringan saraf tiruan, melakukan *forward* dan *backward propagation*, serta mengelola proses *training* dan evaluasi model secara fleksibel (Paszke dkk., 2023). PyTorch juga menyediakan integrasi GPU yang sangat berguna untuk mempercepat proses *training* model yang kompleks (Babichev 2020).

7. tqdm

   *Library* tqdm digunakan untuk memberikan progres bar yang informatif selama proses pelatihan model. Dalam penelitian ini, tqdm mempermudah pemantauan proses pelatihan cGAN dan BERT dengan menampilkan estimasi waktu dan kecepatan iterasi setiap epoch. Hal ini sangat membantu dalam debugging dan evaluasi performa selama eksperimen berlangsung, khususnya pada proses training yang cukup memerlukan banyak waktu (da Costa-Luis 2022).

8. *Hugging Face Transformers*

   *Transformer*s dari *Hugging Face* adalah *library* yang menyediakan model *pre-trained* seperti BERT dan IndoBERT. Dalam penelitian ini, *library* ini digunakan untuk melakukan tokenisasi teks menggunakan *Tokenizer* dan melakukan *fine-tuning* terhadap *BertForSequenceClassification* (Wolf dkk., 2020). Keunggulan *transformers* terletak pada kemudahan akses terhadap model bahasa canggih yang sudah dilatih dengan data besar, sehingga dapat langsung digunakan atau dilatih ulang sesuai kebutuhan penelitian (Sun dkk., 2020).

9. *Scikit-learn* (sklearn)

   *Scikit-learn* digunakan dalam penelitian ini untuk berbagai kebutuhan pembelajaran mesin yang tidak secara langsung ditangani oleh deep learning. Beberapa fungsi utama dari *scikit-learn* yang digunakan antara lain

*LabelEncoder* untuk mengubah label teks menjadi numerik, *TfidfVectorizer* untuk mengekstrak fitur teks berbasis TF-IDF sebelum digunakan oleh model cGAN, serta evaluasi model menggunakan metrik seperti *classification report*, *confusion matrix*, dan *train-test split* (Barupal dan Fiehn 2021).

10. *Streamlit*

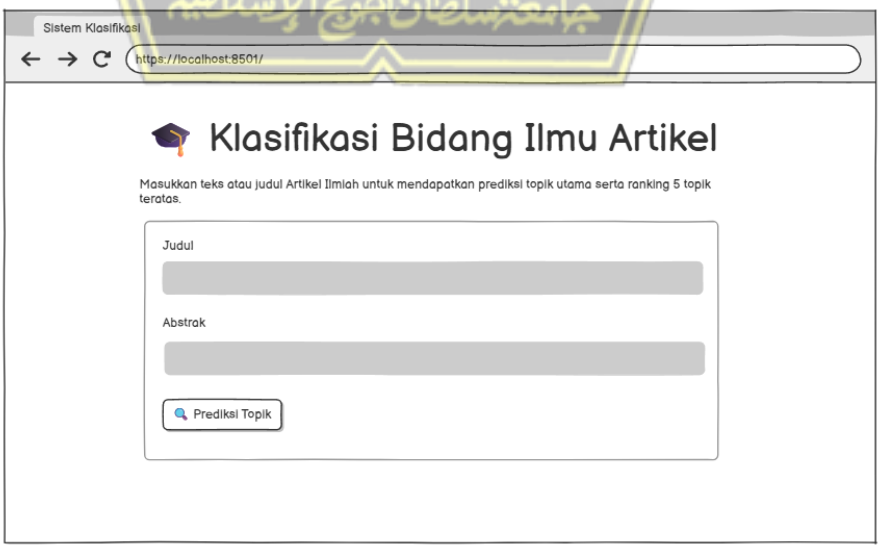
*Stremlit* merupakan salah satu *framework library open source* berbasis *python* yang mendukung *deployment* model ke dalam program berbasis *website* (Deshpande dkk., 2025). Streamlit digunakan untuk membangun antarmuka pengguna interaktif secara cepat, sederhana, dan efisien, khususnya untuk aplikasi *machine learning* dan *data science* (Muller dkk., 2025). Dalam penelitian ini *streamlit* digunakan untuk mengimplementasikan sistem klasifikasi bidang ilmu ilmiah berbasis BERT yang telah dilatih sebelumnya sehingga pengguna dapat langsung memasukkan judul dan abstrak jurnal yang diteliti untuk mendapatkan prediksi topik secara *realtime*.

## 3.4 Perancangan *User Interface*

Berikut ini merupakan rancangan desain dari sistem yang akan diterapkan dalam penelitian ini:

1. Halaman Utama

Halaman utama ini adalah desain antarmuka yang akan menjadi tampilan pertama kali dilihat oleh pengguna.

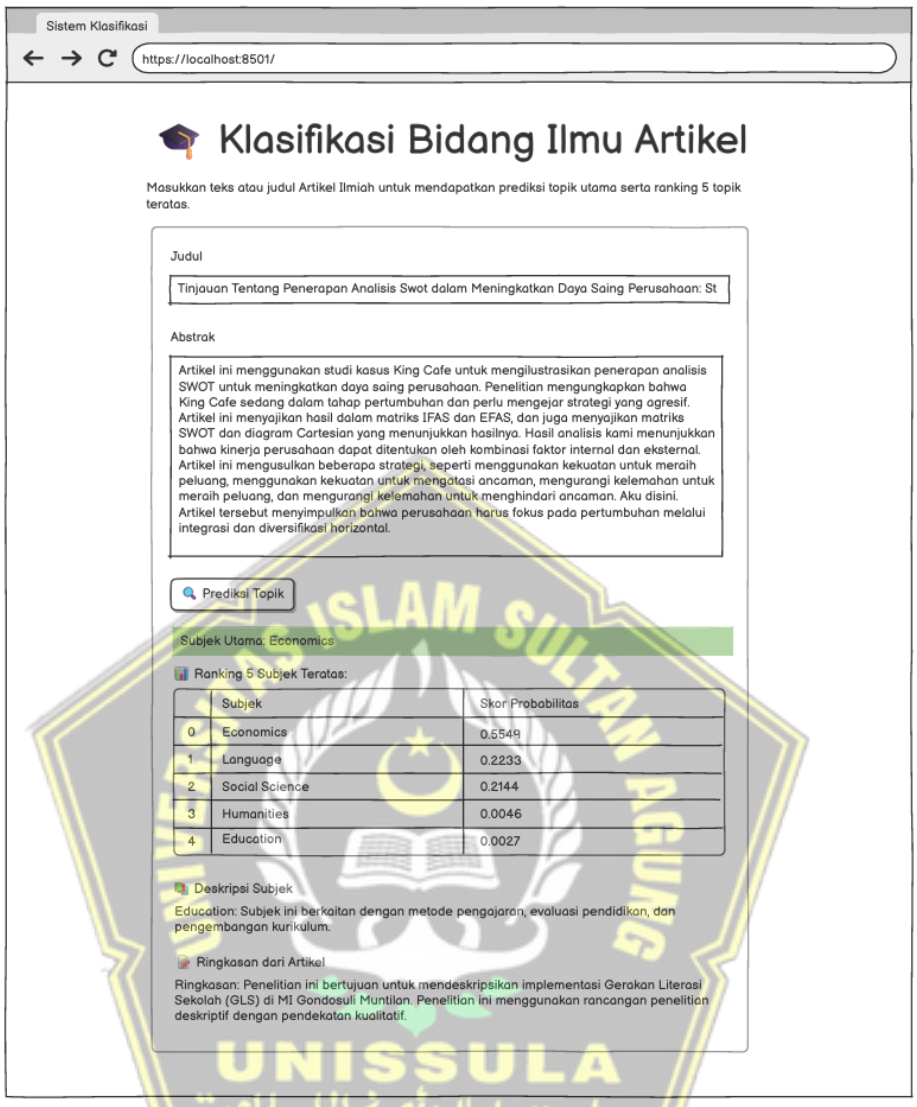



Gambar 2. 14 Halaman Utama

Gambar 3. 10 merupakan halaman tampilan utama dari sistem klasifikasi artikel ilmiah terindeks Garuda ketika pengguna awal mengakses *website* tersebut. Pada halaman ini pengguna akan melihat tampilan *title* bertuliskan "Klasifikasi Bidang Ilmu Artikel", yang dibawahnya terdapat teks instruksi yang memerintahkan untuk memasukkan judul dan abstrak dari artikel yang akan di klasifikasikan, kemudian dibawahnya terdapat tombol "Prediksi Topik" dimana data akan dikirimkan *input* ke sistem agar dapat diproses dan menghasilkan *output* berupa hasil prediksi bidang ilmu yang paling relevan dengan data yang dimasukkan.

2. Halaman Prediksi

Halaman prediksi merupakan halaman yang menampilkan hasil dari proses prediksi teks artikel ilmiah yang dimasukkan oleh pengguna, seperti terlihat pada gambar 3. 11 di bawah ini.




Gambar 2. 15 Halaman Prediksi

Gambar 3. 11 merupakan halaman sistem klasifikasi yang menampilkan hasil dari proses prediksi berdasarkan judul dan abstrak artikel ilmiah yang dimasukkan oleh pengguna. Terdapat hasil prediksi subjek utama yang paling relevan dengan *input* teks, kemudian terdapat tabel yang menampilkan *ranking* 5 subjek teratas dengan 2 kolom yaitu kolom subjek dan kolom skor probabilitas. Data akan diurutkan dari skor tertinggi dan akan menjadi subjek utama yang berada di nomor 1 dan seterusnya hingga ke nomor 5 dengan bidang ilmu yang memiliki skor probabilitas paling kecil. Kemudian terdapat kolom "Deskripsi Subjek" yang berisikan deskripsi singkat artikel, juga terdapat kolom "Ringkasan dari Artikel" yang berisi ringkasan dari *input*an artikel.

# BAB IV
# HASIL DAN ANALISIS PENELITIAN

## 4.1 Hasil Penelitian

Penelitian ini menghasilkan sebuah sistem klasifikasi bidang ilmu artikel ilmiah terindeks Garuda menggunakan metode cGAN – BERT. Proses penelitian diawali dengan mengumpulkan dataset yang bersumber dari *platform* Garuda untuk diambil judul dan abstrak dari artikel ilmiah dengan 5 bidang ilmu yaitu *Economics*, *Education*, *Humanities*, *Social Science*, dan *Language*.

Setelah dilakukan pembersihan dan standarisasi data, ditemukan adanya ketimpangan distribusi antar bidang ilmu yaitu dengan jumlah keseluruhan 4226 data dengan rincian *Economics* 1142 data, *Language* 769 data, *Social Science* 874 data, *Humanities* 730 data, *Education* 711 data yang berarti jumlah data tidak seimbang satu sama lain dan dapat mengakibatkan sistem tidak belajar dengan baik dan akan terjadi bias pada salah satu bidang ilmu dengan jumlah data yang lebih banyak.

Untuk mengatasi ketidakseimbangan tersebut, diterapkan metode cGAN guna menghasilkan data sintetik yang menyerupai data asli. Proses training cGAN dilakukan selama 5000 *epoch* dengan menggunakan representasi fitur dari TF-IDF. Selanjutnya, seluruh data (asli dan sintetik) digunakan untuk melatih model IndoBERT (*indobenchmark/indobert-base*-p1) dalam melakukan klasifikasi. Pelatihan dilakukan selama 5 *epoch* menggunakan 80% data sebagai *training* dan 20% sebagai *validation*.

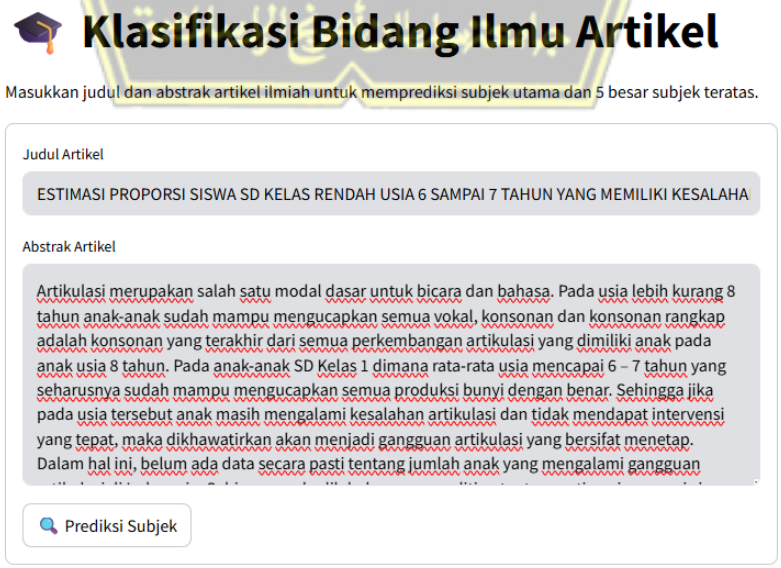
Setelah model berhasil dijalankan, langkah selanjutnya adalah mengimplementasikan aplikasi dalam bentuk *website*. *Website* ini dirancang agar pengguna dapat langsung memasukkan teks judul dan abstrak artikel ilmiah yang akan diteliti, kemudian sistem akan memproses *input* tersebut dan menampilkan hasil prediksi bidang ilmu pada halaman hasil. Dengan memanfaatkan *Streamlit*, aplikasi ini dapat dijalankan secara lokal atau diunggah ke server, sehingga dapat diakses dengan lebih mudah oleh pengguna lain.

### 4.1.1 Tampilan Halaman Utama


Gambar 4. 1 Halaman Utama

Gambar 4. 1 merupakan halaman utama dari *website*, dimana tampilan ini merupakan halaman pertama yang akan dilihat oleh pengguna ketika mengakses sistem. Pada halaman ini, terdapat *form input* perintah untuk memasukkan judul dan abstrak dari artikel yang akan diteliti. Kemudian pengguna dapat menekan tombol "Prediksi Subjek" untuk mengirimkan input tersebut agar dapat diproses oleh sistem. Menjadi catatan penting bahwa kedua kolom *input*an tersebut harus terisi agar tombol "Prediksi Subjek" dapat berfungsi aktif.

### 4.1.2 Halaman *form Input* artikel




Gambar 4. 2 Halaman *form input* artikel

Gambar 4.2 merupakan tampilan halaman ketika pengguna mengisi *form input* judul dan abstrak dari artikel yang akah diteliti. Sistem akan memproses artikel tersebut ketika pengguna mengklik tombol “Prediksi Subjek” yang kemudian akan menghasilkan *feedback* atau balasan prediksi subjek utama dari artikel yang dimasukkan pengguna. Halaman ini menunjukkan bagaimana interaksi awal pengguna dengan sistem sebelum mendapatkan hasil atau respons dari proses yang dijalankan.

### 4.1.3 Halaman Hasil Prediksi


Gambar 4. 3 Contoh hasil prediksi *Humanities*

Gambar 4.3 merupakan *output* sistem yang telah diproses dengan hasil prediksi subjek berdasarkan artikel yang dimasukkan oleh pengguna. Untuk kolom “Subjek Utama: *Humanities*” adalah hasil dari bidang ilmu yang paling relevan dengan artikel dan memiliki skor probabilitas paling tinggi diantara bidang ilmu lainnya sehingga dijadikan sebagai subjek utama dalam sistem prediksi tersebut. Kemudian terdapat tabel yang berisi *ranking* 5 subjek teratas dimana hasil ini diambil dari skor probabilitas dari paling tinggi hingga yang paling rendah. Contoh diatas menghasilkan data *Humanities* dengan skor 0. 9704, *Economics* 0. 0233, *Social sciene* 0. 0049, *Language* 0. 0009, dan *Education* 0. 0005.


| | Subjek | Skor Probabilitas |
|---|---|---|
| 0 | Education | 0.9959 |
| 1 | Economics | 0.0017 |
| 2 | Language | 0.0012 |
| 3 | Humanities | 0.0007 |
| 4 | Social Science | 0.0006 |

Gambar 4. 4 Contoh hasil prediksi *Education*

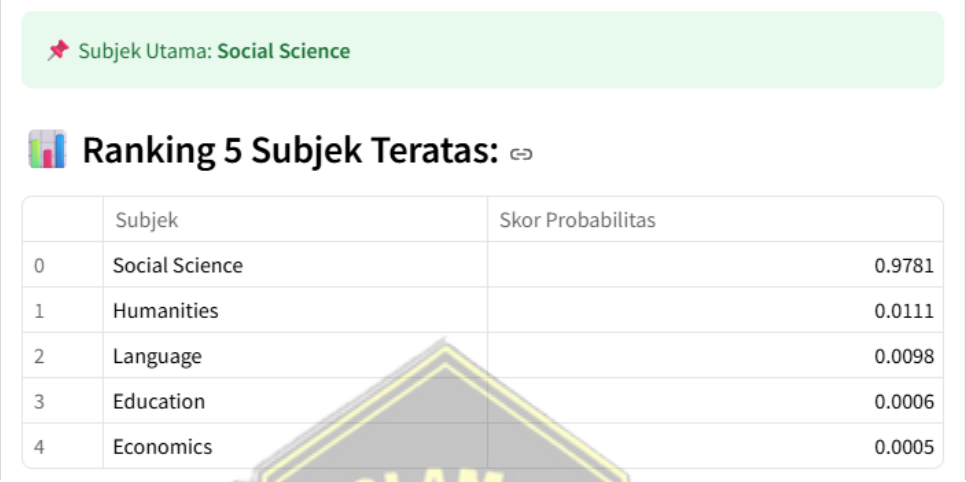
Gambar 4. 4 merupakan contoh dari *output* prediksi artikel yang dimasukkan dengan hasil yang diperoleh yaitu "Subjek Utama: *Education*" yang artinya bidang ilmu *Education* merupakan bidang yang paling relevan dengan artikel yang telah di*input*kan pengguna, kemudian terdapat tabel *ranking* 5 subjek yang menghasilkan data skor probabilitas *Education* 0. 9977, *Social Science* 0. 0010, *Public Health* 0. 0005, *Economics* 0. 0004 , dan *Computer Science* & IT 0. 0003.


| | Subjek | Skor Probabilitas |
|---|---|---|
| 0 | Economics | 0.9863 |
| 1 | Humanities | 0.0110 |
| 2 | Language | 0.0010 |
| 3 | Social Science | 0.0009 |
| 4 | Education | 0.0008 |

Gambar 4. 5 Contoh hasil prediksi *Economics*

Gambar 4. 5 adalah contoh dari *output* prediksi artikel yang dimasukkan dengan hasil yang diperoleh yaitu "Subjek Utama: *Economics*" yang artinya bidang ilmu *Economics* merupakan bidang yang paling relevan dengan artikel yang telah di*input*kan pengguna, kemudian terdapat tabel *ranking* 5 subjek yang

menghasilkan data skor probabilitas *Economics* 0. 9863, *Humanities* 0. 0110, *Language* 0. 0010 , *Social Science* 0. 0009, dan *Education* 0. 0008.




| | Subjek | Skor Probabilitas |
|---|---|---|
| 0 | Social Science | 0.9781 |
| 1 | Humanities | 0.0111 |
| 2 | Language | 0.0098 |
| 3 | Education | 0.0006 |
| 4 | Economics | 0.0005 |

Gambar 4. 6 Contoh hasil prediksi *Social Science*

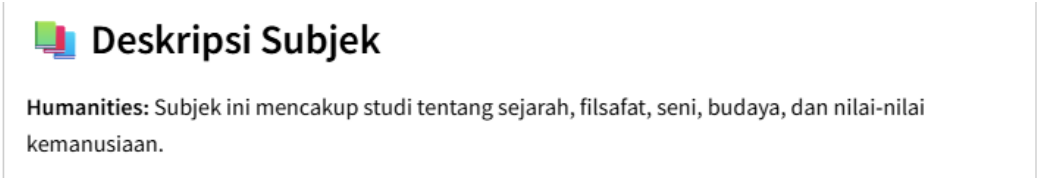
Gambar 4. 6 adalah contoh dari *output* prediksi artikel yang dimasukkan dengan hasil yang diperoleh yaitu "Subjek Utama: *Social Science*" yang artinya bidang ilmu *Social Sciences* merupakan bidang yang paling relevan dengan artikel yang telah di*input*kan pengguna, kemudian terdapat tabel *ranking* 5 subjek yang menghasilkan data skor probabilitas *Social Science* 0. 9781, *Humanities* 0. 0111, *Language* 0. 0098, *Education* 0. 0006, dan *Economics* 0. 0005.


| | Subjek | Skor Probabilitas |
|---|---|---|
| 0 | Language | 0.9914 |
| 1 | Social Science | 0.0054 |
| 2 | Humanities | 0.0014 |
| 3 | Education | 0.0012 |
| 4 | Economics | 0.0007 |

Gambar 4. 7 Contoh hasil prediksi *Language*

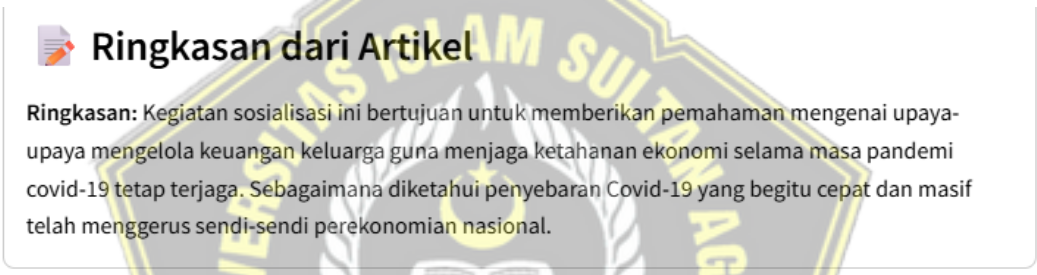
Gambar 4. 7 merupakan contoh dari *output* prediksi artikel yang dimasukkan dengan hasil yang diperoleh yaitu "Subjek Utama: *Language*" yang artinya bidang ilmu *Language* merupakan bidang yang paling relevan dengan artikel yang telah

di*input*kan pengguna, kemudian terdapat tabel *ranking* 5 subjek yang menghasilkan data skor probabilitas *Language* 0. 9914, *Social science* 0. 0054, *Humanities* 0. 0014, *Education* 0. 0012, dan *Economics* 0. 0007.




Gambar 4. 8 deskripsi subjek utama

Gambar 4. 8 merupakan tampilan dari deskripsi subjek utama bidang ilmu artikel. Dimana terdapat penjelasan singkat mengenai deskripsi dari subjek utama yang paling relevan.




Gambar 4. 9 tampilan ringkasan dari artikel

Gambar 4. 9 merupakan tampilan dari halaman "Ringkasan dari Artikel" yang berisikan kalimat yang diambil dari artikel yang dimasukkan oleh pengguna kemudian dilakukan peringkasan kalimat oleh sistem.

## 4.2 Analisa Penelitian

Penelitian ini membangun sistem klasifikasi artikel ilmiah terindeks Garuda menggunakan kombinasi cGAN dan BERT sebagai metode untuk meningkatkan akurasi dan ketepatan dalam mengklasifikasikan artikel pada *platform* Garuda. Fokus utama dalam penelitian ini adalah bagaimana mengatasi ketidakseimbangan distribusi data antar bidang ilmu serta menghasilkan sistem klasifikasi yang adaptif terhadap konteks semantik dari artikel ilmiah.

Dalam prosesnya, langkah awal penelitian ini adalah melakukan *web scraping* dari *platform* Garuda untuk lima bidang ilmu yaitu *Economics, Education, Humanities, Social Science*, dan *Language.* Data yang diperoleh tersebut kemudian

diproses dengan tahap *cleaning*, *case folding*, tokenisasi, serta penghapusan duplikat. Digunakan metode cGAN untuk menghasilkan data sintetik sesuai dengan distribusi bidang yang paling dominan, sehingga data pelatihan menjadi lebih seimbang.

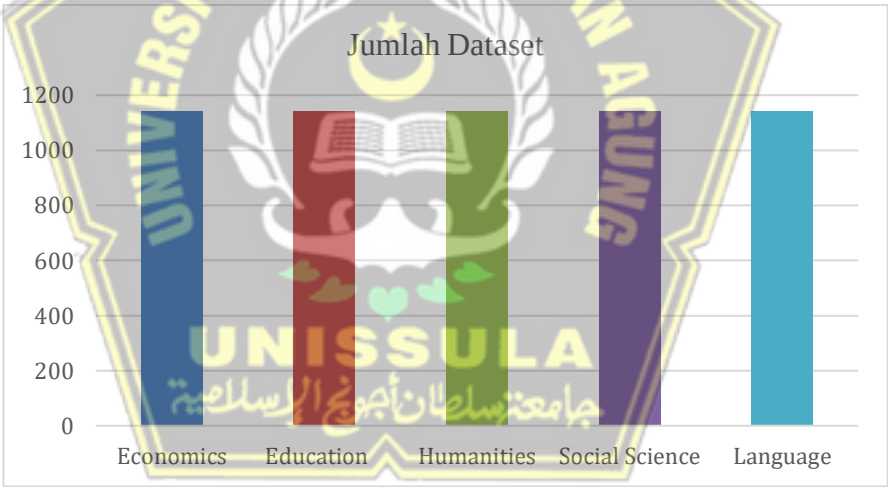
Setelah proses augmentasi dengan cGAN, model IndoBERT dilatih menggunakan data gabungan antara data asli dan data sintetik. Model dilatih selama 5 *epoch* dengan validasi menunjukan akurasi data 87% serta distribusi prediksi yang merata untuk ke-5 bidang ilmu tersebut. Untuk pengujian sistem, dibangun antarmuka berbasis *Streamlit* yang memungkinkan pengguna memasukkan judul dan abstrak artikel ilmiah. Sistem kemudian memprediksi subjek utama, menyajikan top 5 probabilitas klasifikasi, dan memberikan deskripsi serta ringkasan otomatis untuk memperkuat pemahaman pengguna terhadap hasil prediksi. Evaluasi sistem dilakukan dengan menggunakan metrik klasifikasi seperti akurasi, presisi, *recall*, dan F1-*score* yang menunjukan bahwa sistem mampu membedakan konteks antar bidang ilmu dengan baik.

Terdapat beberapa tantangan yang dihadapi dalam proses pengerjaan penelitian ini. Pertama adalah pada saat proses *scraping* data dari *platform* Garuda menghadapi ketidakseragaman format dan kelengkapan metadata, sehingga diperlukan tahap *pre-processing* yang cukup kompleks. Kemudian model cGAN juga cukup sensitif terhadap kualitas data latih dan parameter pelatihan, sehingga diperlukan eksperimen berulang untuk menghasilkan data sintetik yang realistis dan tidak keluar konteks. Tantangan lainnya berhubungan dengan proses *fine-tuning* IndoBERT, dimana proses ini memerlukan komputasi yang cukup tinggi, sementara keterbatasan perangkat keras pada *google colab* memiliki *limit* waktu jika menggunakan *Graphics Processing Unit* (GPU) sehingga menjadi penghambat dalam pelatihan model yang optimal. Selain itu, integrasi sistem ke dalam antarmuka *Streamlit* juga membutuhkan penyesuaian agar mampu memberikan hasil klasifikasi yang tidak hanya akurat, tetapi juga mudah dipahami oleh pengguna akhir. Namun, dengan pendekatan augmentasi melalui cGAN dan pelatihan *fine-tuning* IndoBERT, sistem berhasil menghasilkan performa klasifikasi yang lebih stabil dibanding metode baseline yang tidak menggunakan augmentasi.

Dengan hasil yang diperoleh, penelitian ini menunjukan bahwa kombinasi cGAN dan BERT mampu membentuk sistem klasifikasi bidang ilmu artikel ilmiah yang tidak hanya akurat tetapi juga responsif terhadap konteks linguistik artikel sehingga dapat membantu pengguna untuk menentukan bidang ilmu yang paling relevan dengan artikel ilmiah tersebut.

## 4.3 Hasil cGAN - BERT

Hasil implementasi *Conditional Generative Adversarial Network* (cGAN) dan *Bidirectional Encoder Representations from Transformers* (BERT) dalam penelitian ini menunjukkan bahwa model mampu memberikan prediksi subjek berdasarkan judul dan abstrak dari artikel ilmiah yang dimasukkan oleh pengguna. Model bekerja dengan memanfaatkan pembuatan data sintetik pada proses augmentasi dengan cGAN dan pelatihan *fine-tuning* dengan BERT.




Gambar 4. 10 *chart* dataset augmentasi data

Dari proses augmentasi menggunakan cGAN menunjukan statistik data setelah proses augmentasi menghasilkan jumlah data yang lebih seimbang untuk ke-lima bidang ilmu yang tersedia yaitu menjadi 5710 data dengan rincian *Economics* 1142 data, *Education* 1142 data, *Humanities* 1142 data, *Social Science* 1142 data, dan *Language* 1142 data.

Berikut data sintetik yang berhasil ditambahkan yaitu sejumlah 1484 data sehingga total seluruh data menjadi 5710 data. Di bawah ini merupakan contoh dari data sintetik hasil augmentasi menggunakan cGAN:

Tabel 4. 1 data augmentasi cGAN

| No | Bidang Ilmu | Judul | Abstrak |
|---|---|---|---|
| 1 | *Education* | peningkatan prestasi belajar bahasa indonesia kelas iii melalui pembelajaran tematik dengan metode bercakap-cakap dan bercerita di sd negeri 22 dangin puri | penelitian ini dilakukan di kelas iii sd negeri 22 dangin puri karena rendahnya prestasi belajar siswa dalam mata pelajaran bahasa indonesia. tujuan dari penelitian tindakan kelas ini adalah mengetahui apakah penerapan model pembelajaran tematik dengan metode bercakap-cakap dan bercerita mampu meningkatkan hasil belajar siswa. data dikumpulkan melalui tes hasil belajar dan dianalisis secara deskriptif. hasilnya menunjukkan peningkatan skor dari 64,72 sebelum tindakan, menjadi 66,47 pada siklus i, dan meningkat lagi menjadi 77,76 pada siklus ii. dengan demikian, metode yang diterapkan terbukti mampu meningkatkan prestasi belajar siswa. |
| 2 | *Humanities* | peningkatan kompetensi guru dan siswa dalam penguasaan pemrograman dan jaringan komputer di kota jayapura | program kemitraan masyarakat ini bertujuan untuk mengatasi keterbatasan pengetahuan dasar para guru smk negeri 2 dan smk negeri 3 kota jayapura dalam bidang pemrograman dan |

| No | Bidang Ilmu | Judul | Abstrak |
|---|---|---|---|
| | | | jaringan komputer. solusi yang ditawarkan adalah pendampingan dan pelatihan langsung kepada guru dan siswa mengenai bahasa pemrograman seperti html, php, dan mysql, serta penguatan kompetensi jaringan berbasis ict menggunakan perangkat mikrotik routerboard. pelatihan dilakukan selama lima hari dengan pendekatan praktik interaktif. hasilnya menunjukkan peningkatan kompetensi sebesar 19,8% untuk pelatihan web di smk negeri 3 dan 21,8% untuk jaringan komputer di smk negeri 2 jayapura. |
| 3 | *Social Science* | peningkatan pemahaman hukum lalu lintas bagi remaja untuk menekan angka kecelakaan | organisasi kesehatan dunia (who) yang merupakan bagian dari perserikatan bangsa-bangsa menyatakan bahwa kecelakaan lalu lintas menempati posisi ketiga sebagai penyebab kematian di indonesia setelah penyakit jantung koroner dan tuberkulosis. setiap tahun, sekitar 1,2 juta orang meninggal dunia akibat kecelakaan lalu lintas, |

| No | Bidang Ilmu | Judul | Abstrak |
|---|---|---|---|
| | | | dengan kelompok usia 15-29 tahun menjadi yang paling banyak terkena dampak. sebagian besar korban (73%) adalah remaja laki-laki. faktor utama penyebab kecelakaan ini adalah rendahnya pemahaman remaja terhadap peraturan lalu lintas yang berlaku, seperti yang diatur dalam undang-undang nomor 22 tahun 2009. oleh sebab itu, edukasi mengenai hukum lalu lintas kepada remaja, khususnya pelajar sma, sangat penting untuk menurunkan angka kecelakaan. kegiatan ini dibagi menjadi tiga tahap: persiapan (perizinan dan survei), pelaksanaan (kuisioner sebelum dan sesudah penyuluhan), serta evaluasi (penyusunan laporan). hasilnya menunjukkan peningkatan pemahaman peserta terhadap hukum dan pelanggaran lalu lintas. |
| 4 | *Language* | pergeseran kosakata pertanian dalam bahasa bali dan dampaknya | penelitian ini menyoroti pergeseran kosakata bahasa bali dalam ranah pertanian dan dampaknya terhadap pelestarian |

| No | Bidang Ilmu | Judul | Abstrak |
|---|---|---|---|
| | | terhadap budaya darma pamacul | budaya lokal, khususnya nilai darma pamacul atau kewajiban seorang petani. fenomena ini dilatarbelakangi oleh perubahan praktik bertani yang berdampak pada perubahan bahasa dan budaya. tujuan penelitian adalah memetakan perubahan kosakata pertanian dan menghubungkannya dengan dinamika budaya lokal. menggunakan pendekatan linguistik budaya dan makrosemantik, data dikumpulkan melalui wawancara dengan petani di tabanan dan buleleng. ditemukan bahwa terdapat perubahan pada kosakata terkait peralatan, proses bertani, hingga interaksi sosial. pergeseran ini menyebabkan generasi muda kesulitan memahami metafora dalam bahasa bali pertanian. |

Hasil evaluasi menunjukkan bahwa penggabungan metode *Conditional Generative Adversarial Network* (cGAN) dan *Bidirectional Encoder Representations from Transformers* (BERT) berhasil menciptakan sistem klasifikasi artikel ilmiah terindeks garuda yang tepat dan relevan. Proses augmentasi data melalui cGAN dapat menyeimbangkan distribusi data pada setiap

bidang ilmu dengan menciptakan artikel sintetik yang menyerupai dengan data asli dalam distribusi semantik. Model BERT yang dilatih ulang dengan data augmentasi menunjukkan kinerja yang baik dalam mengklasifikasikan artikel berdasarkan subjek utama seperti *Economics*, *Education, Humanities*, *Social Science*, dan *Language*. Hasil prediksi dari model menunjukkan pemahaman yang kontekstual mengenai judul dan abstrak artikel, dengan probabilitas klasifikasi yang konsisten terhadap isi yang tersedia.

Disamping itu, model dapat secara otomatis memberikan saran subjek lain beserta penjabaran subjek utama berdasarkan *input*an teks yang diberikan oleh pengguna. Kemampuan ini menunjukkan bahwa sistem tidak hanya mengkategorikan artikel ke dalam subjek tertentu secara umum, tetapi juga mampu mengidentifikasi nuansa dan keterkaitan semantik antar subjek. Evaluasi model dengan data validasi menunjukkan akurasi yang tinggi, menandakan bahwa pendekatan cGAN – BERT efektif dalam mengatasi permasalahan ketidakseimbangan data sembari mempertahankan akurasi prediksi.

Secara keseluruhan, penelitian ini membuktikan bahwa metode cGAN – BERT dapat menciptakan sistem klasifikasi artikel ilmiah yang tidak hanya tepat, tetapi juga responsif terhadap variasi subjek di *platform* Garuda. Sistem yang dibuat memiliki peluang untuk ditingkatkan lebih lanjut dalam skala yang lebih besar sebagai bagian dari sistem pencarian dan saran artikel ilmiah berbasis NLP.

## 4.4 Hasil Evaluasi

Evaluasi sistem klasifikasi berguna untuk mengukur performa model BERT setelah proses pelatihan dengan data hasil augmentasi menggunakan cGAN. Evaluasi penting dilakukan untuk menentukan sejauh mana model mampu melakukan tugas klasifikasi dengan benar. Dalam penelitian ini, evaluasi dilakukan menggunakan metrik evaluasi kinerja model seperti akurasi, presisi, *recall*, dan f1-*score*.

Model BERT dievaluasi menggunakan *testing set* yang tidak dilibatkan dalam proses pelatihan maupun augmentasi. Berdasarkan pengujian, model BERT mampu mencapai akurasi sebesar 87%. Hal ini menunjukan bahwa proses augmentasi

berhasil membantu meningkatkan representasi bidang ilmu yang sebelumnya memiliki jumlah data lebih sedikit, seperti *language*, *education*, *social science*, dan *humanities*.

Tabel 4. 2 Hasil Evaluasi Sistem

| **Subject** | **Precision** | **Recall** | **F1-Score** | **Support** |
|---|---|---|---|---|
| Economics | 0.79 | 0.79 | 0.79 | 229 |
| Education | 0.91 | 0.96 | 0.94 | 228 |
| Humanities | 0.81 | 0.79 | 0.80 | 228 |
| Language | 0.93 | 0.93 | 0.93 | 229 |
| Social Science | 0.89 | 0.86 | 0.87 | 228 |
| **Average** | | | | |
| **Accuracy** | | | 0.87 | 1142 |
| **Macro avg** | | | 0.87 | 1142 |
| **Weighted avg** | | | 0.87 | 1142 |

Berdasarkan hasil pengujian menggunakan metrik evaluasi klasifikasi, terlihat adanya variasi kinerja model pada masing-masing bidang ilmu. Secara umum, model menunjukkan performa yang cukup baik dalam mengklasifikasikan artikel jurnal ke dalam 5 bidang yang telah ditentukan. Evaluasi dilakukan dengan membandingkan nilai *precision*, *recall*, dan f1-*score* dari setiap bidang ilmu. Berikut penjelasan rinci mengenai hasil evaluasi tersebut:

1. Subjek *Economics* merupakan bidang ilmu dengan jumlah data tertinggi yang tidak mengalami proses augmentasi, tetapi mencatat kinerja yang cukup baik, dengan nilai presisi, *recall*, dan f1-*score* di angka 0.79 atau 79%. Hal tersebut menandakan model masih melewatkan sejumlah data yang seharusnya dikenali sebagai *Economics* (*false negative*), model tetap membutuhkan variasi representasi untuk menangkap seluruh cakupan dari subjek tersebut.
2. Subjek *Education* mengalami peningkatan yang cukup signifikan setelah augmentasi, dengan nilai *presisi*, *recall*, dan f1-*score* mencapai angka 0.91 atau

91%. Hal ini menunjukan bahwa data tambahan yang diberikan ke model melalu augmentasi mampu meningkatkan representasi dari subjek ini, sehingga model menjadi lebih percaya diri dalam melakukan prediksi. Maka augmentasi pada subjek *Education* terbukti memberikan dampak positif secara nyata terhadap performa klasifikasi.

3. Subjek *Humanities* mencatat nilai presisi 0.81 atau 815, *recall* 0.79 atau 79%, dan f1-*score* 0.80 atau 80%, yang merupakan salah satu nilai terendah dalam laporan ini. Angka tersebut mengindikasikan bahwa model sedikit kesulitan mengklasifikasikan artikel dari bidang ilmu lain sebagai *Humanities* (*false positive*). Kemungkinan konten antar subjek masih memiliki kesamaan kata-kata kunci atau konteks yang menyebabkan model kesulitan dalam membedakan. Hal ini menjadi perhatian karena dapat menurunkan kepercayaan terhadap prediksi model pada subjek ini.
4. Subjek *Language* menunjukan performa tinggi yang sangat konsisten dengan nilai presisi, *recall*, dan f1-*score* mencapai 0.93 atau 93%, yang menandakan proses augmentasi terhadap subjek ini sangat efektif. Angka tersebut menunjukkan bahwa model mampu mengenali artikel dengan subjek *Language* secara akurat dan konsisten. Hal ini dapat disebabkan oleh keberagaman data hasil augmentasi serta kekonsistenan pola Bahasa pada subjek ini yang lebih mudah ditangkap oleh model BERT.
5. Subjek *Social Science* mencatat hasil yang cukup baik dengan nilai presisi 0.89 atau 89%, *recall* 0.86 atau 86%, dan f1-*score* 0.87 atau 87%. Angka ini mengindikasikan bahwa augmentasi berhasil memberikan variasi data yang membantu model memahami konteks Social Science, meskipun masih ada ruang untuk perbaikan dalam membedakan kategori ini dari yang lain.

Berdasarkan hasil evaluasi terhadap 1142 data uji, model menghasilkan akurasi keseluruhan sebesar 0.87 atau 87%. Akurasi ini menunjukkan tingkat kesesuaian prediksi model terhadap label sebenarnya dalam data uji. Selain akurasi, nilai *macro average* dan *weighted average* jug mencapai 0.87 atau 87%.

Angka *macro average* menandakan jika model mempunyai performa yang seimbang terhadap semua kelas tanpa memandang jumlah data di tiap subjek

tersebut, sedangkan *weighted average* memperhitungkan proporsi jumlah data pada setiap subjek. Hal ini membuktikan bahwa proses augmentasi dengan cGAN dan proses *embeding* teks menggunakan BERT berhasil meningkatkan distribusi data untuk kelas minoritas dan performa untuk sistem klasifikasi bidang ilmu artikel ilmiah terindeks Garuda dengan tepat dan relevan.

# BAB V
# KESIMPULAN DAN SARAN

## 5.1 Kesimpulan

Berdasarkan hasil penelitian ini, dapat diambil kesimpulan bahwa sistem klasifikasi bidang ilmu artikel ilmiah terindeks Garuda dapat berjalan dengan baik, dengan didukung teknik augmentasi data menggunakan *Conditional Generative Adversarial Network* (cGAN) dan teknik *text embeding* untuk memahami konteks teks menggunakan *Bidirectional Encoder Representations from Transformers* (BERT). Permasalahan ketidak seimbangan data antar subjek berhasil di atasi dengan melakukan augmentasi pada subjek-subjek minoritas, seperti *education*, *humanities*, *social science*, dan *language*, sementara subjek *economics* dikecualikan karena jumlah data dianggap telah mencukupi.

Pendekatan ini memungkinkan sistem untuk menghasilkan distribusi data yang lebih seimbang. Ini menghasilkan proses klasifikasi yang lebih stabil dan tidak lagi bias terhadap topik mayoritas. Hasil evaluasi menunjukkan bahwa metrik klasifikasi seperti akurasi, presisi, *recall*, dan f1-*score* dapat ditingkatkan dengan menggunakan data hasil peningkatan, terutama berlaku untuk bidang ilmu yang sebelumnya memiliki data yang tidak seimbang. Selain itu, model BERT sangat baik dalam memahami konteks teks jurnal dan mengklasifikasikannya ke dalam subjek yang relevan. Oleh karena itu, integrasi antara augmentasi berbasis cGAN dan model BERT terbukti efektif dalam meningkatkan akurasi sistem klasifikasi bidang ilmu artikel ilmiah dan menyelesaikan masalah yang sering terjadi dalam pemrosesan data tidak seimbang yang berkaitan dengan tugas klasifikasi teks.

## 5.2 Saran

Berdasarkan hasil dalam penelitian ini, penulis memberikan beberapa saran yang dapat dijadikan masukan untuk pengembangan lebih lanjut, diantaranya yaitu:

1. Optimasi proses training cGAN untuk menghindari *overfitting* dan menghasilkan vektor fitur yang lebih beragam dan tidak terlalu mirip dengan data asli.

2. Perluasan cakupan bidang ilmu jurnal dan penelitian label multilabel, karena banyak artikel ilmiah yang mencakup lebih dari satu bidang, membuat model lebih fleksibel dalam pengklasifikasian artikel.
3. Penelitian ini melakukan eksperimen dengan IndoBERT sebagai arsitektur *transformator* tambahan untuk model klasifikasi, penelitian selanjutnya dapat mencoba arsitektur transformator yang lebih besar atau disesuaikan khusus untuk klasifikasi teks di domain ilmiah seperti XLM-RoBERTa, IndoBERTweet, atau BigBird.
4. Banyak jurnal di Indonesia yang menggunakan campuran bahasa Inggris. Untuk menangani variasi bahasa ini dan meningkatkan kekuatan sistem klasifikasi, penelitian lanjutan dapat mencoba pendekatan multibahasa.

# DAFTAR PUSTAKA


Adhikari, Ashutosh, Achyudh Ram, Raphael Tang, and Jimmy Lin. 2020. "DocBERT: BERT for Document Classification." http://arxiv.org/abs/1904.08398.

Alqulaity, Malak, and Po Yang. 2024. "Enhanced Conditional GAN for High-Quality Synthetic Tabular Data Generation in Mobile-Based Cardiovascular Healthcare." *Sensors* 24(23): 1–20.

Anggraeni, Diah Bekti, Widyastuti Widyastuti, Fitri Puji Rahmawati, and Madya Giri Aditama. 2021. "Pengembangan Sistem Klasifikasi Kepustakaan Dengan Dewey Decimal Classification (DDC)." *Buletin KKN Pendidikan* 3(2): 152–60.

Anisatuzzumara. 2024a. "Implementasi Latent Dirichlet Allocation (LDA) Dan K-Nearest Neighbors(KNN) Pada Sistem Rekomendasi Jurnal Terindeks GARUDA." *Αγαη* 15(1): 37–48.

———. 2024b. 15 "Implementasi Latent Dirichlet Allocation (LDA) Dan K-Nearest Neighbors (KNN) Pada Sistem Rekomendasi Jurnal Terindeks Garuda." Universitas Islam Sultan Agung.

Babichev, Eugeny. 2020. "Emergence of Ghosts in Horndeski Theory." *Journal of High Energy Physics* 2020(7).

Barupal, Dinesh Kumar, and Oliver Fiehn. 2021. "Generating the Blood Exposome Database Using a Comprehensive Text Mining and Database Fusion Approach." *Environmental Health Perspectives* 127(9): 2825–30.

Bhat, Ranjith, and Raghu Nanjundegowda. 2025. "A Review on Comparative Analysis of Generative Adversarial Networks' Architectures and Applications." *Journal of Robotics and Control (JRC)* 6(1): 53–64.

Carneiro, Tiago et al. 2020. "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications." *IEEE Access* 6: 61677–85.

Chandra, Abel, Laura Tünnermann, Tommy Löfstedt, and Regina Gratz. 2023. "Based Deep Learning for Predicting Protein Properties in the Life Sciences." :

1–25.

Comparison, Learning. 2022. “Detection of Abnormal SIP Signaling Patterns : A Deep.” : 1–17.

da Costa-Luis, Casper O. 2022. “Tqdm: A Fast, Extensible Progress Meter for Python and CLI.” *Journal of Open Source Software* 4(37): 1277.

Croce, Danilo, Giuseppe Castellucci, and Roberto Basili. 2020. “GAN-BERT: Generative Adversarial Learning for Robust Text Classification with a Bunch of Labeled Examples.” *Proceedings of the Annual Meeting of the Association for Computational Linguistics*: 2114–19.

Cutting, Vineesh, and Nehemiah Stephen. 2021. “A Review on Using Python as a Preferred Programming Language for Beginners.” *International Research Journal of Engineering and Technology* 8(8): 4258–63. www.irjet.net.

Dameani, Tiara. 2021. “Analisis Panel Data Web Scraping Artikel Kekerasan Dalam Rumah Tangga Tahun 2019- 2020 Di DKI Jakarta.” *Jurnal Teknologi Informasi* 7(1): 43–49.

Deshpande, Paras et al. 2025. “Automated Result Analysis Using Python and Streamlit.” *International Journal of Data Science and Analytics* 3(1): 1–20.

Dhruv, Akshit J., Reema Patel, and Nishant Doshi. 2021. “Python: The Most Advanced Programming Language for Computer Science Applications.” (Cesit 2020): 292–99.

Hafiz, Y. A., and Endah Sudarmilah. 2023. “Implementasi Web Scraping Pada Portal Berita Online.” *Inisiasi*: 55–60.

Hajkowicz, Stefan et al. 2023. “Artificial Intelligence Adoption in the Physical Sciences, Natural Sciences, Life Sciences, Social Sciences and the Arts and Humanities: A Bibliometric Analysis of Research Publications from 1960-2021.” *Technology in Society* 74.

Islam, Saidul et al. 2023. “A C OMPREHENSIVE S URVEY ON A PPLICATIONS OF.”

Kenton, Ming-wei Chang, Lee Kristina, and Jacob Devlin. 2022. “BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding.” (Mlm).

Kuang, Kun et al. 2021. "BertGCN: Transductive Text Classification by Combining GCN and BERT." *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*: 1456–62.

Li, Yang et al. 2024. "Recent Developments in Recommender Systems: A Survey [Review Article]." *IEEE Computational Intelligence Magazine* 19(2): 78–95.

Ma, Lijing, and Shiru Qu. 2022. "Application of Conditional Generative Adversarial Network To." (March 2021).

Muller, Tom David et al. 2025. "OpenMS WebApps: Building User-Friendly Solutions for MS Analysis." *Journal of Proteome Research* 24(2): 940–48.

Nayla, Adine, Casi Setianingsih, and Burhanuddin Dirgantoro. 2023. "Deteksi Hate Speech Pada Twitter." *eProceeding of Engineering* 10(1): 256.

Oktafiandi, H. 2023. "Implementasi LDA Untuk Pengelompokan Topik Twitter Bertagar# Mypertamina." *Jurnal Ekonomi dan Teknik Informatika* 11(1): 10–16. https://www.e-journal.polsa.ac.id/index.php/jneti/article/view/222%0Ahttps://www.e-journal.polsa.ac.id/index.php/jneti/article/download/222/154.

Pan, Zhaoqing et al. 2021. "Recent Progress on Generative Adversarial Networks (GANs): A Survey." *IEEE Access* 7: 36322–33.

Paszke, Adam et al. 2023. "PyTorch: An Imperative Style, High-Performance Deep Learning Library." *Advances in Neural Information Processing Systems* 32(NeurIPS).

Rask, Jonas Kjær et al. 2021. "Visual Studio Code VDM Support." *Proceedings of the 18th International Overture Workshop* (December): 35–50.

Ribas, Lucas C., Wallace Casaca, and Ricardo T. Fares. 2025. "Conditional Generative Adversarial Networks and Deep Learning Data Augmentation: A Multi-Perspective Data-Driven Survey Across Multiple Application Fields and Classification Architectures." *AI (Switzerland)* 6(2).

Rogers, Anna, Olga Kovaleva, and Anna Rumshisky. 2020. "A Primer in Bertology: What We Know about How Bert Works." *Transactions of the Association for Computational Linguistics* 8: 842–66.

Sa'adah, Farikhatus. 2022a. "Klasifikasi Bidang Ilmu Pada Publikasi Terindeks

Garuda Menggunakan Metode K-Nearest Neighbor (K-Nn).” *Angewandte Chemie International Edition, 6(11), 951–952*. 5(2): 5–24. http://repo.iain-tulungagung.ac.id/5510/5/BAB 2.pdf.

———. 2022b. Angewandte Chemie International Edition, 6(11), 951–952. “Klasifikasi Bidang Ilmu Pada Publikasi Terindeks Garuda Menggunakan Metode K-Nearest Neighbor (K-Nn).” Universitas Islam Sultan Agung. http://repo.iain-tulungagung.ac.id/5510/5/BAB 2.pdf.

Shah, Momna Ali, Muhammad Javed Iqbal, Neelum Noreen, and Iftikhar Ahmed. 2023. “An Automated Text Document Classification Framework Using BERT.” *International Journal of Advanced Computer Science and Applications* 14(3): 279–85.

Sharma, Vijeta, Gaurav Kumar Gupta, and Manjari Gupta. 2021. “Performance Benchmarking of GPU and TPU on Google Colaboratory for Convolutional Neural Network.” (May): 639–46.

Snehkunj, Rupal, Khushboo Vachiyatwala, and Corresponding Author. 2022. “Data Analysis Using Pandas Library of Python.” *Acta Scientific COMPUTER SCIENCES* 4(3): 37–41. https://actascientific.com/ASCS/pdf/ASCS-04-0236.pdf.

Sun, Chi, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2020. “How to Fine-Tune BERT for Text Classification?” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 11856 LNAI(2): 194–206.

Supardi, Cholid Fajar. 2023. “Final Project Trend Search System Final Project Title of Unissula Informatics Engineering Students Using Keyword Extraction.” Universitas Islam Sultan Agung.

Suprapti, Tati et al. 2023. “Implementasi Model Algoritma Generative Adversarial Network (Gan) Pada Sistem Presensi Berbasis Deteksi Wajah (SIDEWA).” *Tematik* 9(2): 231–36.

Syarifudin, Faisal. 2022. “Klasifikasi Artikel-Artikel Jurnal Pustakaloka Berdasarkan Skema Jita.” *Fihris: Jurnal Ilmu Perpustakaan dan Informasi* 17(1): 20.

Vaswani, Ashish et al. 2017. “Attention Is All You Need.” *Advances in Neural Information Processing Systems* 2017-December(Nips): 5999–6009.

Van Der Walt, Stéfan, S. Chris Colbert, and Gaël Varoquaux. 2021. “The NumPy Array: A Structure for Efficient Numerical Computation.” *Computing in Science and Engineering* 13(2): 22–30.

Wang, Zhengwei, Qi She, and Tomás E. Ward. 2021. “Generative Adversarial Networks in Computer Vision: A Survey and Taxonomy.” *ACM Computing Surveys* 54(2).

Widiansyah, Muhammad, Fathia Frazna Az-zahra, and Agung Pambudi. 2021. “Fine-Tuning Model Indobert ( Indonesian Bidirectional Encoder Representations from Transformers ) Untuk Analisis Sentimen Berbasis Aspek Pada Aplikasi M-Paspor.”

Wijaya, Bhianta, and Edi Surya Negara. 2022. “Penerapan Garuda Smart City Model Dalam Menganalisa Kesiapan Pemerintah Kabupaten Tulang Bawang Barat Dalam Membangun Konsep Smart City.” *CogITo Smart Journal* 8(2): 524–36.

Wolf, Thomas et al. 2020. “Transformers: State-of-the-Art Natural Language Processing.” *EMNLP 2020 - Conference on Empirical Methods in Natural Language Processing, Proceedings of Systems Demonstrations*: 38–45.