

BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

2.1.1. Chatbot

Chatbot atau *chatterbot* adalah sebuah program komputer yang diciptakan untuk meniru percakapan manusia, baik melalui suara maupun teks (Astari et al., 2023). *Chatbot* menjadi alat layanan pelanggan yang handal karena kemampuan mereka untuk menyimpan banyak data tanpa mengabaikannya dan kemampuan mereka untuk langsung bertanya kepada sumber data daripada mencari sendiri. *Chatbot* dapat digunakan di banyak bidang, seperti bisnis, pendidikan, hiburan, dan layanan publik. Asisten virtual seperti Amazon Alexa dan Google Assistant adalah contoh *chatbot* (Raharjo, 2023).

2.1.2. Kitab Fathul Qarib

Fathul Qorib adalah salah satu kitab yang menjadi referensi utama untuk mempelajari bidang fikih di berbagai pesantren Indonesia (Hasbi et al., 2022). *Fathul Qorib*, karya ulama asal Palestina, Syamsuddin Abu Abdillah Muhammad bin Qosim al Ghazi, tidak hanya mengajarkan ilmu fikih. Buku ini juga bertujuan membentuk karakter santri dengan menanamkan nilai-nilai seperti syukur, pengendalian diri, dan semangat kebersamaan. Oleh karena itu, kitab ini sangat digemari oleh pondok pesantren di seluruh Nusantara

2.1.3. Natural Language Processing

Natural Language Processing yaitu salah satu bidang studi yang berfokus pada bagaimana komputer dapat memahami, memahami, dan menanggapi bahasa

manusia (Chandra et al., 2020). Pengolahan bahasa alami membantu pengguna dan komputer berkomunikasi dalam bahasa murni. Jika pengguna tidak memiliki cukup waktu untuk belajar atau menguasai bahasa baru, proses pengolahan bahasa (NLP) akan membantu.

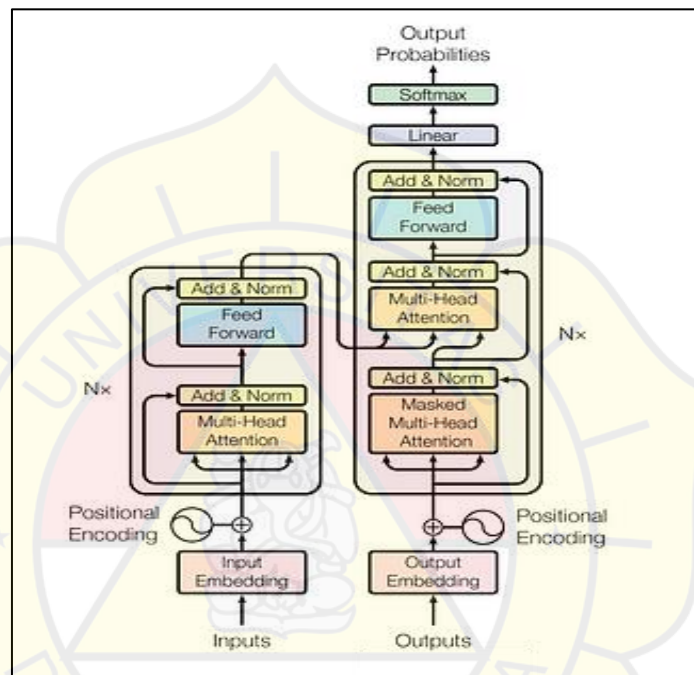
2.1.4. Deep Learning

Deep Learning adalah bagian dari kecerdasan buatan yang menggunakan jaringan saraf tiruan berlapis untuk melakukan tugas-tugas seperti mengenali gambar, suara, dan menerjemahkan bahasa dengan sangat akurat (Raup et al., 2022). *Deep learning* adalah sebuah teknik dalam jaringan saraf tiruan yang terdiri dari berbagai lapisan untuk memfasilitasi pemahaman representasi data dengan berbagai tingkat abstraksi. Dalam pembelajaran mendalam, lapisan terdiri dari tiga komponen: lapisan input, lapisan tersembunyi, dan lapisan output. Pada *deep learning*, lapisan tersembunyi dibentuk secara bertingkat untuk menemukan pola yang tepat dan mengurangi kesalahan pada hasil akhir. Deep learning memungkinkan mesin belajar langsung dari data seperti gambar, teks, atau suara tanpa perlu aturan yang terlalu spesifik. Proses pembelajaran ini bisa dilakukan dengan cara diawasi, semi-diawasi, atau tanpa pengawasan.

2.1.5. Transformer

Transformer telah merevolusi bidang pemrosesan bahasa alami dengan kemampuannya dalam menangkap hubungan antar kata dalam kalimat secara lebih baik dibandingkan model sebelumnya. Arsitektur encoder-decoder yang digunakan Transformer memungkinkan model untuk menghasilkan output yang lebih koheren dan relevan dengan inputnya. Mekanisme perhatian yang diterapkan pada setiap lapisan memungkinkan model untuk memfokuskan pada bagian-bagian penting

dari input, sehingga menghasilkan representasi yang lebih informatif. Di sisi lain, setiap lapisan dalam *decoder* memanfaatkan semua representasi yang dihasilkan oleh *encoder* untuk menghasilkan urutan keluaran dengan menggunakan informasi kontekstual yang telah digabungkan. Gambar 2.1 adalah penjelasan tentang arsitektur *Transformer*.



Gambar 2. 1 Arsiktektur Transformer (Bahari & Dewi, 2024)

Encoder: Encoder terdiri dari beberapa lapisan identik yang masing-masing mengubah input awal menjadi representasi internal yang mengandung informasi kontekstual. Setiap lapisan encoder terdiri dari dua sub-lapisan utama: mekanisme self-attention dan feed-forward neural network. Mekanisme self-attention memungkinkan setiap posisi dalam input untuk mempertimbangkan informasi dari semua posisi lainnya, membantu model memahami hubungan antar kata dalam kalimat. Setelah self-attention, hasilnya diteruskan ke feed-forward neural network yang menerapkan transformasi non-linear. Setiap sub-lapisan ini diikuti oleh

operasi normalisasi lapisan dan koneksi residual untuk meningkatkan stabilitas dan efisiensi pelatihan. Input diproses dari satu lapisan ke lapisan berikutnya, menghasilkan representasi yang semakin kaya dan mendalam.

Decoder: Decoder juga terdiri dari beberapa lapisan identik yang menggunakan representasi internal dari encoder untuk menghasilkan urutan keluaran yang diinginkan. Setiap lapisan decoder memiliki tiga sub-lapisan utama: *masked self-attention*, *encoder-decoder attention*, dan *feed-forward neural network*. *Masked self-attention* memastikan bahwa setiap posisi dalam keluaran hanya dapat mempertimbangkan posisi sebelumnya, menjaga urutan yang benar dalam generasi teks. *Encoder-decoder attention* memungkinkan decoder untuk menggunakan representasi yang dihasilkan oleh encoder untuk memperkaya proses *decoding*. Seperti pada *encoder*, setiap sub-lapisan di *decoder* diikuti oleh normalisasi lapisan dan koneksi residual. Decoder memproses input dari lapisan ke lapisan hingga menghasilkan urutan keluaran akhir yang akurat dan kontekstual.

2.1.6. Bidirectional Encoder Representation from Transformer

BERT adalah model kecerdasan buatan yang sangat bagus dalam memahami bahasa manusia. Model ini dilatih dengan banyak data teks sehingga bisa memahami konteks kata dalam kalimat dengan sangat baik. (Afifa et al., 2023). BERT, singkatan dari Bidirectional Encoder Representations from Transformers, adalah model pembelajaran mendalam yang dirancang untuk memahami hubungan antara kata-kata dalam sebuah kalimat. Model ini menggunakan arsitektur Transformer yang memungkinkan BERT untuk memproses seluruh kalimat sekaligus, baik dari kiri ke kanan maupun sebaliknya. Berbeda dengan model sebelumnya yang hanya mempertimbangkan konteks satu arah, BERT mampu

menangkap informasi kontekstual yang lebih kaya, sehingga menghasilkan representasi kata yang lebih akurat. BERT dilatih pada kumpulan data teks yang sangat besar untuk mempelajari pola bahasa yang umum. Setelah dilatih, model BERT yang sudah jadi dapat dengan mudah disesuaikan (fine-tuning) untuk tugas-tugas spesifik seperti klasifikasi teks atau menjawab pertanyaan. Hal ini membuat BERT menjadi model yang sangat efisien dan efektif untuk berbagai aplikasi.

BERT dapat digunakan untuk pre-training dan fine-tuning. Selama pre-training, model dilatih untuk menyelesaikan berbagai tugas pre-training menggunakan data yang tidak berlabel. Selama fine-tuning, model BERT dimulai dengan parameter yang telah dilatih sebelumnya dan kemudian diubah dengan data yang memiliki label untuk menyelesaikan berbagai tugas proses pengolahan bahasa natural (NLP), seperti mengklasifikasikan teks, menjawab pertanyaan, dan mengidentifikasi entitas bernama.

1. Pre-training

Pretraining adalah tahap awal dalam pengembangan model bahasa di mana model dilatih pada korpus teks yang sangat besar dan beragam. Tujuannya adalah untuk membuat model mempelajari representasi kata dan kalimat yang umum dan bermakna. Model yang telah dilatih sebelumnya ini kemudian dapat diadaptasi dengan mudah untuk tugas-tugas spesifik dengan sedikit data tambahan melalui proses fine-tuning.

2. Fine-tuning

Dengan menambahkan lapisan yang sesuai dengan tugas yang ingin diselesaikan pada model yang sudah ada, model BERT yang telah dilatih sebelumnya dapat digunakan untuk menangani berbagai tugas NLP lainnya.

BERT dapat dioptimalkan untuk berbagai tugas natural language, termasuk sentiment analysis, jawaban pertanyaan, pengenalan entitas bernama, dan klasifikasi teks.

2.1.7. IndoBERT

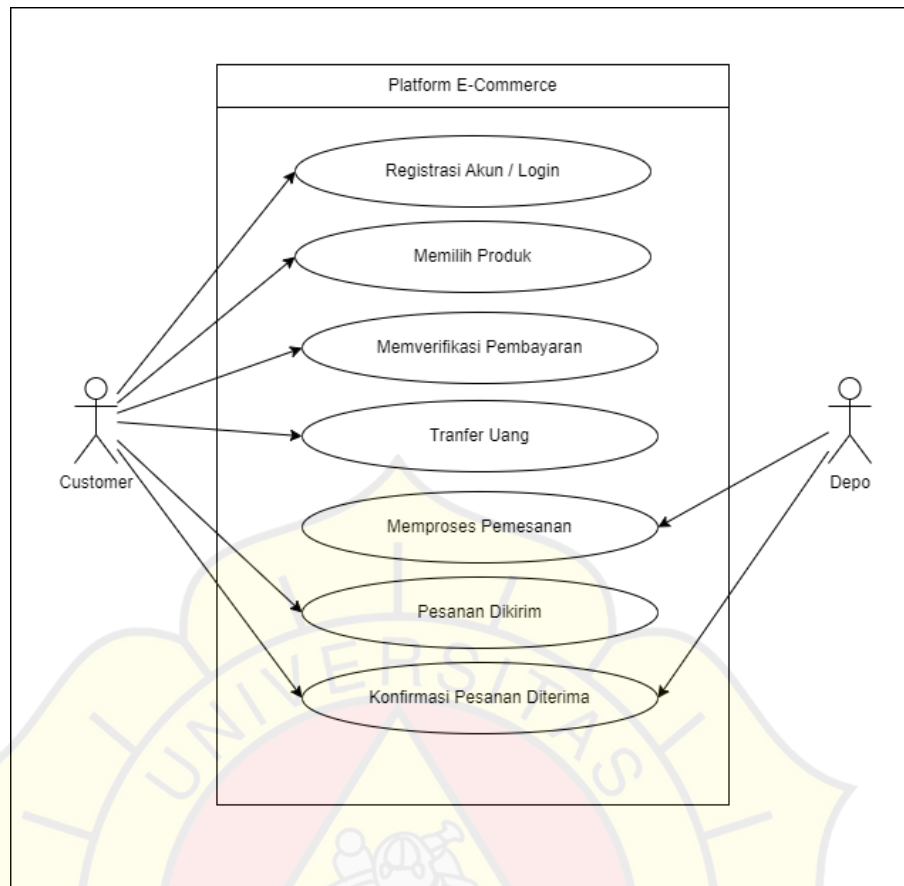
IndoBERT adalah model BERT khusus yang dilatih menggunakan data Bahasa Indonesia. Model ini mempunyai 4 versi yang berbeda yaitu IndoBERT_{BASE}, IndoBERT_{LARGE}, dan dua sisanya berbasis ALBERT, yaitu IndoBERT-lite_{BASE}, IndoBERT-lite_{LARGE}. Data yang digunakan oleh IndoBERT diambil dari Wikipedia Indonesia (74 Juta kata), Artikel berita dari Kompas, Tempo dan Liputan6 (total 55 juta kata), Corpus Web Indonesia memiliki 90 juta kata (Koto et al., 2020).

2.1.8. Unified Modeling Language

Unified Modeling Language (UML) adalah salah satu *tool* atau *model* untuk merancang pengembangan software yang berbasis *object-oriented* (Sonata, 2019). Selain itu, UML sendiri menyediakan standar penulisan sistem *blueprint*. Standar ini mencakup konsep proses bisnis, skema database, penulisan kelas dalam bahasa program tertentu, dan bagian yang diperlukan untuk sistem software.

A. Use Case Diagram

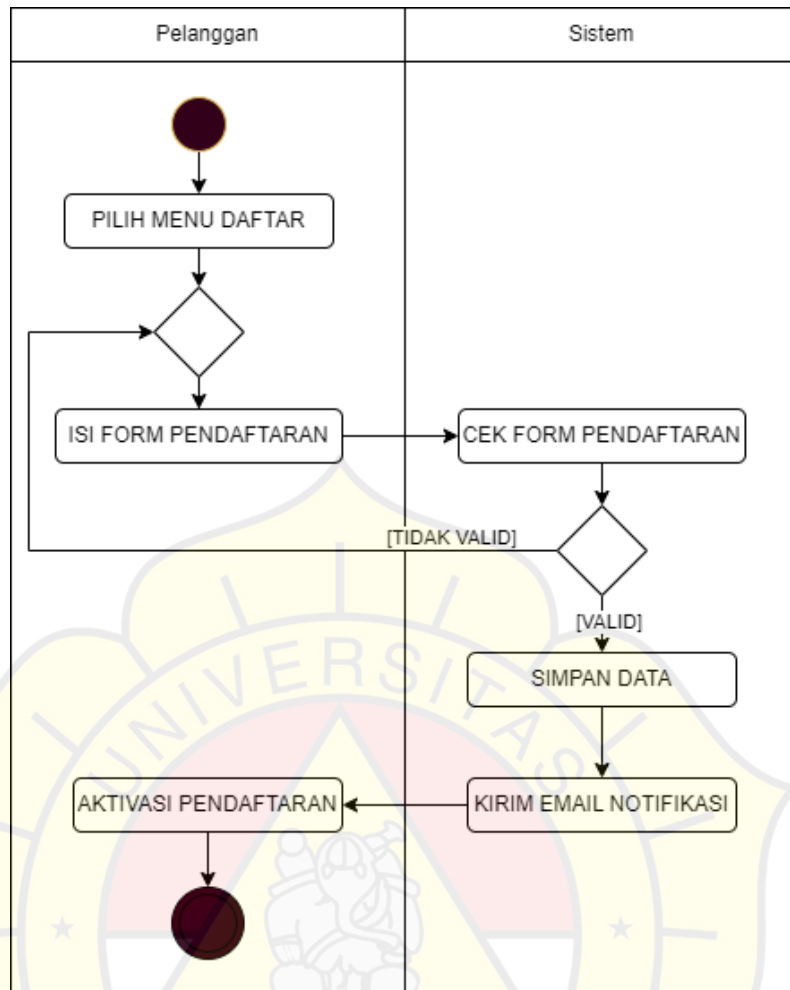
Use Case Diagram menunjukkan sistem dari sudut pandang pengguna dan menggambarkan interaksi antara aktor yang terlibat dalam sistem informasi. Contohnya seperti gambar 2.2.



Gambar 2. 2 Contoh Use Case Diagram (Sonata, 2019)

B. Activity Diagram

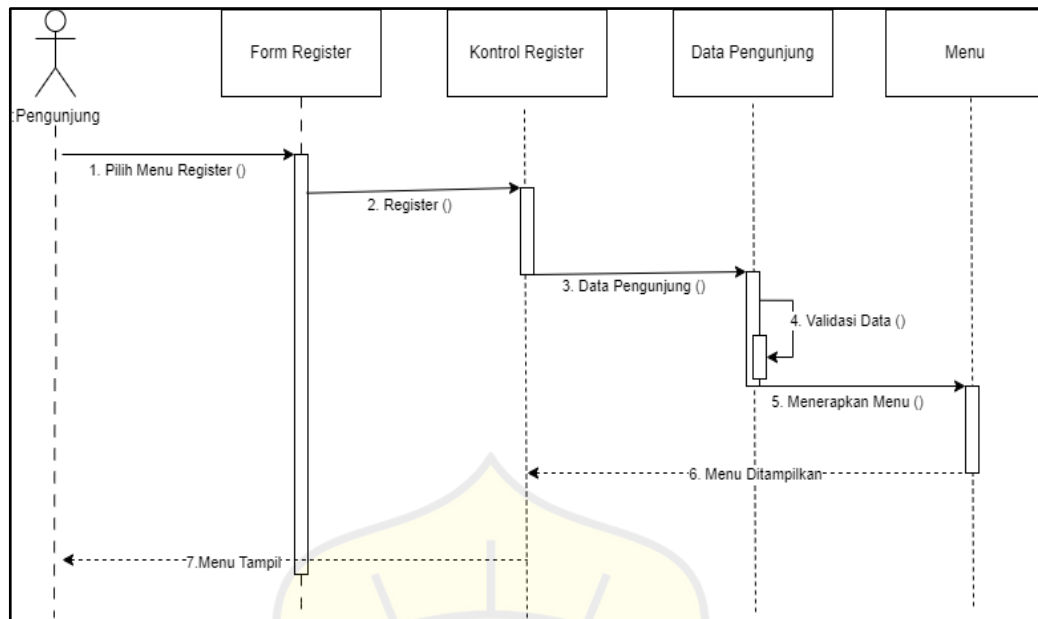
Activity Diagram adalah representasi grafis yang menggambarkan perilaku objek terpisah selama proses bisnis. Mereka dapat digunakan untuk memodelkan berbagai tingkat proses bisnis, mulai dari alur kerja bisnis yang lebih umum dengan berbagai kasus penggunaan hingga rincian kasus penggunaan per individu. Contohnya seperti gambar 2.3.



Gambar 2. 3 Contoh Activity Diagram (Sonata, 2019)

C. Sequence Diagram

Diagram Urutan (*Sequence Diagram*) berfokus pada perilaku dalam sistem dengan menggambarkan bagaimana objek saling berinteraksi. Diagram ini menampilkan objek dan pesan yang dikirim di antara objek-objek tersebut. Umumnya, diagram urutan digunakan untuk menunjukkan interaksi objek yang terjadi dalam suatu use case. Contoh dari sequence diagram pada gambar 2.4.



Gambar 2. 4 Contoh Sequence Diagram (Sonata, 2019)

2.1.9. CRISP-DM (Cross Industry Standart Process for Data Mining)

CRISP-DM adalah kerangka kerja yang telah diakui secara luas dalam industri data mining. Kerangka kerja ini terdiri dari enam tahap yang saling berhubungan, mulai dari pemahaman bisnis hingga penerapan model. CRISP-DM memberikan struktur yang sistematis untuk proyek data mining, memastikan bahwa setiap langkah dilakukan secara menyeluruh.

Proses data mining berdasarkan CRISP-DM (Cross Industry Standard Process for Data Mining) terdiri dari enam fase, adalah sebagai berikut:

1. *Business Understanding Phase* (Fase Pemahaman Bisnis)

Tahap pertama adalah memahami tujuan bisnis dan persyaratan dari proyek. Ini melibatkan pengumpulan informasi tentang masalah bisnis yang ingin diselesaikan, menetapkan tujuan, dan mendefinisikan bagaimana hasil dari data mining akan memberikan nilai bisnis.

2. *Data Understanding Phase* (Fase Pemahaman Data)

Pada tahap ini, data yang relevan dikumpulkan dan dieksplorasi. Ini mencakup penilaian awal terhadap data untuk memahami kualitas, karakteristik, dan potensi masalah. Analisis ini membantu dalam menentukan bagaimana data akan digunakan dalam proses mining.

3. *Data Preparation Phase* (Fase Persiapan Data)

Data preparation melibatkan pembersihan dan transformasi data agar siap untuk analisis lebih lanjut. Ini termasuk penanganan data yang hilang, konversi format data, penggabungan dataset, dan penghapusan duplikasi.

4. *Modelling Phase* (Fase Pemodelan)

Di tahap ini, teknik data mining yang sesuai dipilih dan diterapkan pada data untuk membangun model. Ini mencakup pemilihan algoritma, pelatihan model, dan evaluasi performa model. Proses ini sering melibatkan eksperimen dengan berbagai teknik untuk menemukan model yang paling efektif.

5. *Evaluation Phase* (Fase Evaluasi)

Setelah model dibangun, tahap ini melibatkan penilaian hasil model terhadap tujuan bisnis yang telah ditetapkan. Evaluasi meliputi analisis hasil, verifikasi apakah model memenuhi kebutuhan bisnis, dan penentuan apakah model siap untuk implementasi.

6. *Deployment Phase* (Fase penyebaran)

Tahap terakhir adalah implementasi model dalam lingkungan bisnis nyata. Ini mencakup penerapan model ke dalam sistem produksi, serta pemantauan dan pemeliharaan untuk memastikan model terus berfungsi dengan baik dan memberikan manfaat yang diharapkan.

2.1.10. Google Colab

Google Colab, juga dikenal sebagai *Google Colaboratory*, adalah layanan berbasis cloud yang memungkinkan pengguna menulis, menjalankan, dan berbagi kode *Python* melalui web browser. Selain itu, *Google Colab* adalah layanan *Jupyter Notebook* yang dapat diakses tanpa persiapan dan memberikan akses gratis ke sumber daya komputasi, termasuk GPU dan TPU.

2.1.11. Python

Python adalah salah satu bahasa pemrograman yang paling banyak digunakan oleh para developer dan perusahaan besar untuk membuat aplikasi berbasis desktop, web, dan mobile. Interpreternya memungkinkan penerjemahan kode secara langsung. (Muhammad Romzi & Kurniawan, 2020). Python menggunakan paradigma pemrograman dari sejumlah bahasa lain; ini termasuk paradigma pemrograman prosedural seperti bahasa C, pemrograman berorientasi objek seperti Java. Kombinasi paradigma ini membantu *software engineer* mengembangkan berbagai proyek dengan Python.

2.2. Kajian Penelitian Terdahulu

Bagian ini menggambarkan hasil penelitian sebelumnya yang terkait dengan topik penelitian saat ini. Penelitian sebelumnya dipilih berdasarkan relevansinya dengan karakteristik penelitian yang sedang dilakukan, dengan maksud memberikan pemahaman yang lebih mendalam dan menyediakan referensi yang berharga bagi penulis selama proses penelitian. Tabel 2.1 di bawah menunjukkan hasil dari penelitian terdahulu.

Tabel 2. 1 Tabel Penelitian Terdahulu

No	Judul	Metode	Hasil
1	Implementasi Natural Language Processing (NLP) Dalam Pengembangan Aplikasi Chatbot Pada SMK YPE Nusantara Slawi (Apriliani et al., 2023)	NLP	Penelitian ini menerapkan Natural Language Processing (NLP) pada aplikasi chatbot di SMK YPE Nusantara Slawi. Chatbot dirancang untuk mempermudah akses informasi sekolah bagi calon siswa dan masyarakat, serta meningkatkan pengalaman pengguna dalam memperoleh informasi yang diperlukan. Dengan pendekatan NLP, informasi kompleks mengenai sekolah, jurusan, guru, dan mata pelajaran dapat disajikan dengan cara yang ringkas dan mudah dipahami, sehingga meningkatkan efisiensi penyampaian informasi. Hasil penelitian menunjukkan bahwa penggunaan NLP dalam pengembangan chatbot membawa beberapa manfaat positif, seperti ekstraksi entitas yang lebih baik, peningkatan pemahaman mengenai sekolah, interaksi real-time dengan pengguna, respons yang relevan, dan aksesibilitas informasi yang lebih baik.

2	Pendekatan Natural Language Processing Pada Aplikasi Chatbot Sebagai Alat Bantu Customer Service (Mulyatun et al., 2021)	NLP dengan Fuzzy String Matching	Penelitian ini membahas tentang pendekatan Natural Language Processing dengan metode Fuzzy String Matching untuk membuat aplikasi Chatbot alat bantu customer untuk memperoleh informasi akademik di Universitas Amikom Yogyakarta. Dalam penelitian ini diharapkan dapat memberikan sebuah informasi tentang akademik kepada mahasiswa dengan cepat dan tanpa ada keterbatasan waktu dan jumlah customer service.
3	Perancangan Chatbot Sebagai Pembelajaran Dasar Bahasa Jawa Menggunakan Artificial Intelligence Markup Language (Abilowo et al., 2020)	Artificial Intelligence Markup Language	Dalam penelitian ini, aplikasi Chatbot yang menawarkan materi dalam tiga tingkatan bahasa Jawa dan mengajukan pertanyaan kepada pengguna dalam tiga tingkatan bahasa Jawa dibahas. Hasil penelitian ini menunjukkan bahwa sistem chatbot dapat membantu mempertahankan bahasa Jawa. Dengan akurasi sebesar 90%, sistem ini mungkin berhasil.