Jurnal Ilmu Komputer dan Informatika (JIKI)
Vol. 3, No. 1, Juni 2023, Hal. 53-64
https://jiki.jurnal-id.com
DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591

Aplikasi Chatbot untuk Layanan Akademik Menggunakan *Platform RASA Open Source* dengan Fitur *Two Stage Fallback*

Nicholas Cannavaro[*1]

[1]Jurusan Informatika, Universitas Tanjungpura, Indonesia
Email: [1]nicholascannavaro@student.untan.ac.id

Abstrak

Sejak dunia dilanda wabah *COVID-19*, jumlah penggunaan *chatbot* sebagai tenaga bantu *customer service* meningkat pesat sehingga memicu pengembangan *chatbot* yang semakin luas dan melahirkan banyak *platform* pihak ketiga untuk memudahkan pengembangan dan menghemat waktu serta biaya, salah satunya adalah *RASA Open Source* dimana dengan pemanfaatan *platform* ini beserta fitur-fiturnya, dapat dihasilkan konsep awal *chatbot* layanan akademik dan analisis dari penelitian yang bisa dijadikan acuan untuk pengembangan berkelanjutan. Penelitian menggunakan metode *CRISP-DM* yang dipadukan dengan model *RAD* untuk pengembangan *chatbot* yang terdiri dari banyak iterasi. Penelitian pada iterasi ke-1 memperoleh *dataset* awal dan di *training* oleh *pipeline model* setelah dilakukan *hyperparameter tuning*, kemudian dilakukan *cross validation* dimana rata-rata akurasi dari kedua proses ini berhasil mencapai nilai 90% keatas. Model *chatbot* yang didapat dari iterasi ke-1 di *deploy* untuk diuji oleh pengguna dan diperoleh 227 sampel baru untuk diuji klasifikasi *intent* dengan nilai akurasi total mencapai 72,02% pada seluruh *intent* dan 76,39% tanpa *intent* percakapan alami. Pengembangan pada iterasi ke-2 diperoleh *dataset* terbaru hasil pembelajaran sampel pengguna di pengujian klasifikasi *intent* awal dan kemudian dilakukan *hyperparameter tuning* serta *cross validation* dengan rata-rata akurasi mengalami penurunan sekitar 1-2% dibanding pada iterasi ke-1. Model terbaru di *deploy* untuk dilakukan kembali pengujian oleh pengguna dan diperoleh 302 sampel baru untuk dilakukan pengujian klasifikasi *intent* dengan nilai akurasi total mencapai 78,40% pada seluruh *intent* dan 82,49% tanpa *intent* percakapan alami.

Kata kunci: *Chatbot, Cross Validation, Dataset, Intent, Hyperparameter Tuning*

Abstract

Since the spread of COVID-19 around the world, the amount of chatbot usage as an aid for customer service has increased rapidly which triggers chatbot development to even wider fields and gives birth to many third platforms to simplify the development and reduce the time and cost for making a chatbot, one of them is RASA Open Source where by utilizing its features, it can produce an initial concept of an academic services chatbot and the analysis from the research that can be used as a reference for continuous development. This research used the CRISP-DM method combined with the RAD model for repetition chatbot development. The first iteration of the research produces an initial dataset that will be trained by a model pipeline after the hyperparameter tuning process and then goes to the cross validation process where the average accuracy from both processes can exceed 90%. The model from the initial development will be deployed for testing by users which able to obtained 227 new samples for intent classification testing with total accuracy of 72,02% on all intents and 76,39% without natural conversation intents. The second iteration was able to form a new dataset from the result of users' samples learning on initial intent classification testing and then goes the hyperparameter tuning with cross validation process which both accumulatively got a decrease on the average of accuracy by 1-2% compared to the first iteration. The new model will be deployed for another testing by users which gained 302 new samples for intent classification testing with total accuracy of 78,40% on all intents and 82,49% without natural conversation intents.

Keywords: *Chatbot, Cross Validation, Dataset, Intent, Hyperparameter Tuning*

DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591

1. PENDAHULUAN

Banyak bidang industri yang mulai memasuki proses transformasi digital termasuk dengan pemanfaatan kecerdasan buatan untuk memudahkan pekerjaan manusia [1], terutama sejak dunia dilanda wabah *COVID-19* pada tahun 2020 sehingga menuntut tiap industri untuk menciptakan sistem kerja dan layanan secara daring dimana salah satu contoh penerapannya adalah dengan menerapkan layanan *chatbot* yang memungkinkan konsumen untuk mendapatkan informasi tanpa perlu melakukan kontak fisik dan dapat meringankan pekerjaan *customer service* [2].

Pengembangan *chatbot* yang semakin meluas membuat banyaknya *platform* pihak ketiga yang lahir dengan tujuan untuk memudahkan pengembangan dan menghemat waktu serta biaya [3]. Untuk *platform* yang digunakan dalam penelitian ini adalah *RASA Open Source*. Setiap *platform* memiliki kelebihan dan kekurangannya masing-masing dari segi kinerja, biaya, dan apakah bersifat *open source* atau tidak, *RASA Open Source* sendiri menjadi pilihan utama para pengembang dikarenakan bersifat *open source* dan tidak memerlukan biaya untuk penggunaannya serta kinerjanya tidak jauh berbeda dengan *platform* berbayar seperti Dialogflow dari Google [4].

Setelah proses pelatihan dan pengujian model *chatbot*, pengembang mendapatkan hasil analisa untuk memastikan apakah model bekerja dengan baik atau tidak [5] namun untuk bisa memperoleh hasil analisa yang spesifik pada pelatihan dan pengujian akan memerlukan implementasi yang cukup rumit apabila tidak didukung dengan pemanfaatan *platform* khusus untuk permasalahan pengembangan *chatbot* [6].

Dalam prakteknya bahkan setelah *chatbot* siap digunakan dan dirilis untuk publik, terkadang masih bisa ditemukan kasus dimana *chatbot* tidak mampu mengklasifikasi dengan pasti pertanyaan pengguna meskipun topik pertanyaannya sudah sesuai dengan ilmu yang dipelajari oleh *chatbot* [7]. Oleh karena itu diperlukan langkah antisipasi untuk memastikan pengguna masih bisa mendapatkan jawaban yang diinginkan dari *chatbot* dengan mengatur langkah kedua yang dapat diambil oleh *chatbot* apabila tidak dapat mengklasifikasi pertanyaan awal dengan nilai probabilitas yang tinggi (*Two stage fallback*) [8].

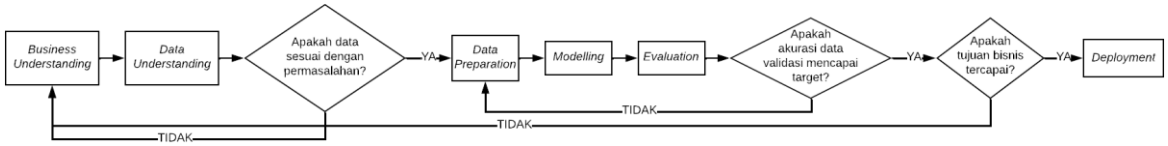
Ukuran *dataset* yang digunakan untuk melakukan pelatihan model *chatbot* sangat berpengaruh, semakin besar ukurannya maka model yang dihasilkan lebih memuaskan hasil klasifikasinya dikarenakan *feature* yang terdapat pada *dataset* lebih banyak [9]. Salah satu cara yang dapat digunakan untuk menambah ukuran *dataset* lagi setelah dibuat sebuah *dataset* awal yang digunakan pada *chatbot* adalah dengan menambahkan data pertanyaan baru yang di-*input* langsung oleh pengguna pada *chatbot* yang telah dibagikan dan harus dirancang agar tidak sampai membocorkan identitas pengirim pertanyaan. Hal ini tidak hanya menambah ukuran *dataset* awal yang telah dirancang oleh pengembang, namun juga dapat membuat *dataset* yang digunakan menjadi lebih natural karena berisi data pertanyaan yang dihasilkan dari percakapan dunia nyata antara pengguna dengan *chatbot* [10].

Chatbot yang dibangun menggunakan *RASA Open Source* berupa *chatbot* level 2 yang dapat menjawab pertanyaan pengguna yang relevan dengan *dataset FAQ*, tidak seperti seorang *customer service* yang bisa menanyakan balik penggunanya mengenai berbagai hal [11]. *Dataset* awal dibuat hingga berukuran minimal 600 sampel yang bersumber dari buku pedoman akademik dan *website* Jurusan Informatika Universitas Tanjungpura dimana kemudian *dataset* diatur kedalam beberapa *file* konfigurasi *project chatbot RASA* hingga kemudian dilakukan proses *hyperparameter tuning* hingga *cross validation* untuk mendapatkan hasil analisa agar dapat men-*deploy chatbot* untuk digunakan pengguna dan mendapat hasil yang baik untuk kemudian diulang lagi seluruh proses ini di iterasi kedua.

2. METODE PENELITIAN

Penelitian akan menggunakan metode *CRISP-DM* (*Cross-Industry Standard Process for Data mining*) yang merupakan standar metode dalam industri *data mining* [12] dan akan dibarengi dengan model *RAD* (*Rapid Application Development*) agar dapat dilakukan pengembangan *chatbot* secara berulang hingga iterasi kedua dengan metode yang menjadi lebih fleksibel setelah iterasi pertama.

DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591



Gambar 1. Alur penelitian mengikuti metode *CRISP-DM*

Berdasarkan Gambar 1, metode *CRISP-DM* akan dimulai dari tahap *business understanding* untuk memahami apa manfaat yang diinginkan dari *chatbot*, *data understanding* untuk memahami informasi apa saja yang perlu dipelajari *chatbot* serta *data preparation* untuk menyiapkan *dataset* yang akan digunakan dalam *training* model, kemudian tahap *modelling* untuk mengatur *pipeline* yang digunakan untuk *training* model, dan hasil dari *training* model setelah dilakukan *hyperparameter tuning* dan hasil dari *cross validation* akan didapatkan pada tahap *evaluation*, barulah *chatbot* kemudian akan di *deploy* ke *platform cloud* untuk diuji secara langsung kepada pengguna dalam tahap *deployment*.

2.1. *Business Understanding*

Pada tahap *business understanding* akan dilakukan riset untuk mengetahui arah tujuan dari diciptakannya *chatbot*. Dalam hal sederhananya *chatbot* akan diciptakan untuk mewujudkan layanan akademik kampus yang dapat diakses dimana saja dan kapanpun kita inginkan, yang mana secara otomatis dapat mempercepat penyebaran informasi penting dan umum mengenai kegiatan akademik yang ada di kampus.

2.2. *Data Understanding*

Kemudian pada tahap *data understanding* dilakukan penelitian terhadap data-data apa saja yang akan diolah untuk dijadikan informasi yang tertampung dalam sebuah *dataset* yang akan digunakan oleh *chatbot* sesuai dengan tujuan bisnis yang sudah ditentukan, dan juga dilakukan penentuan kategori informasi apa saja yang akan digunakan serta memastikan kualitas informasi yang akan dimasukkan ke dalam *dataset* agar informasi yang tersampaikan ke pengguna dapat dipahami dengan baik. Pemahaman terhadap data yang digunakan perlu ditekankan dengan baik agar tidak melenceng dari tujuan bisnis.

2.3. *Data Preparation*

Pada tahap *data preparation* dilakukan pengecekan kembali pada data yang akan digunakan untuk memastikan tidak adanya masalah yang ditemukan, misalnya ada kejadian dimana 1 contoh pertanyaan bisa berada pada 2 kategori yang memiliki kandungan informasi yang sangat berbeda dan akan mengakibatkan hasil akurasi yang buruk pada 2 kategori tersebut apabila tidak diatasi dan jumlah contoh pertanyaan yang salah tempatnya banyak.

Dataset yang telah dirancang pada tahap *data understanding* memiliki berbagai macam kelas *intent*(maksud) dengan respon yang diberikan untuk pelatihan model *chatbot* layanan akademik kampus. Kelas-kelas *intent* yang dirancang juga termasuk kelas yang menangani percakapan sederhana seperti ‘salam’, ‘sampai jumpa’, dan juga tersedia *intent* ‘diluar informasi’ yang berfungsi untuk menyaring masukan dari pengguna yang tidak berhubungan dengan tema informasi yang dipelajari oleh *chatbot*.

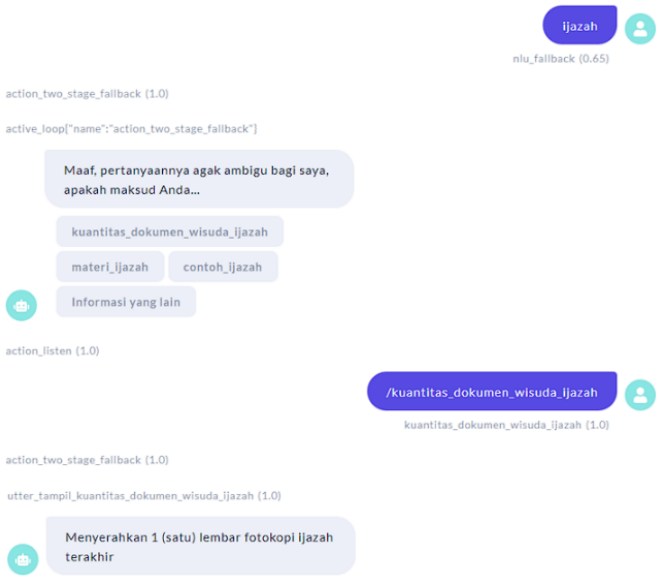
2.4. *Modelling*

Kemudian pada tahap *modelling* akan dilakukan perancangan model *chatbot* untuk menentukan algoritma dan *hyperparameter* yang digunakan pada tiap fase, dimana pada model *NLP* (*Natural Language Processing*) yang digunakan pada *chatbot* terutama pada *platform RASA Open Source* biasanya terdiri mulai dari fase *Tokenization*, *Featurization*, *Intent Classification*, *Entity Extraction*,

DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591

Response Selector, serta *batch size* dan *learning rate* yang akan digunakan, dan *Fallback Classifier* yang termasuk juga dalam mengatur fitur *Two stage fallback*.

Chatbot akan dikembangkan pada level 2 (*FAQ Assistants*) yang akan menjawab pertanyaan pengguna dalam satu tahap. Itu adalah konsep sederhana dari *chatbot FAQ Assistants*, namun dalam penelitian ini akan dilakukan penerapan pengambilan tahap kedua oleh *chatbot* apabila pertanyaan pengguna yang diklasifikasi memiliki nilai probabilitas yang rendah, dimana pengguna akan disediakan beberapa pilihan kategori informasi yang memiliki tingkat probabilitas tertinggi sebagai informasi yang diinginkan pengguna. Inilah yang bisa diwujudkan dengan pemanfaatan fitur *Two stage fallback* yang dapat dilihat simulasi penggunaannya pada Gambar 2.



Gambar 2. Simulasi *Two stage fallback*

Pada Gambar 2, pengguna mencoba *input* kata ’ijazah’ yang kemudian memicu langkah pertama dari *Two stage fallback* dikarenakan memiliki nilai probabilitas tertinggi dibawah *threshold* yang diatur, dan *chatbot* menawarkan beberapa jenis informasi seperti pada gambar yang dapat diklik untuk langsung mendapatkan jawaban. Pengguna dapat mengklik tombol ’Informasi yang lain’ apabila tidak ada jenis informasi yang diinginkan untuk kemudian pengguna memasuki langkah kedua/terakhir dari *Two stage fallback* dimana pengguna akan mencoba menanyakan *chatbot* dengan kalimat yang lebih jelas, dan apabila pengguna masih belum dapat memperoleh jawaban yang diinginkan, maka *chatbot* akan mengakhiri percakapan tersebut.

2.5. *Evaluation*

Pada tahap *evaluation* akan dilakukan evaluasi pada hasil *hyperparameter tuning* model *chatbot* dan setelah mendapatkan *hyperparameter* yang optimal akan dilakukan pengujian model *chatbot* dengan menggunakan *dataset* yang telah dibuat sebelumnya dimana hasil pengujian data validasi yang menggunakan metode *cross validation* akan dicek tingkat akurasinya berdasarkan nilai rata-rata komponen berikut di tiap kelas *intent*:

a. *Precision*

Precision adalah nilai keakuratan hasil klasifikasi tiap kalimat pesan dari *dataset* yang terklasifikasi ke satu kelas *intent*, dimana setiap kalimat yang terklasifikasi terhadap satu kelas *intent* akan ada yang sesuai dengan labelnya (*True Positive/TP)* dan yang tidak sesuai (*False Positive/FP*), sehingga didapatkan rumus *precision* seperti berikut.

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591

b. *Recall*

Recall adalah nilai keakuratan hasil klasifikasi tiap kalimat pesan dari *dataset* yang berasal dari kelas *intent* yang sama, dimana terdapat kalimat yang hasil klasifikasinya benar (*True Positive/TP)* dan yang terklasifikasi ke kelas *intent* yang lain (*False Negative/FN*), sehingga didapatkan rumus *recall* seperti berikut.

$$Recall = \frac{TP}{TP+FN} \qquad (2)$$

c. *F1-score*

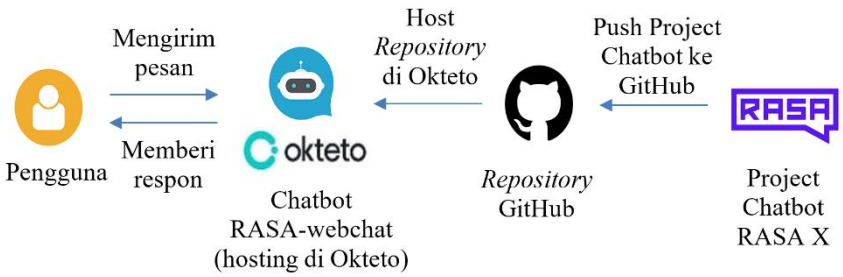
F1-score adalah nilai kombinasi dari *precision* dan *recall* yang digunakan untuk menambah *variasi* nilai keakuratan dari klasifikasi *intent* oleh *chatbot* dengan rumus seperti berikut.

$$F1 - score = \frac{2*(P*R)}{P+R} \; (P = Precision, R = Recall) \qquad (3)$$

Nilai *precision*, *recall*, dan *f1-score* dapat diperoleh menggunakan fitur-fitur analisa yang terdapat pada *platform RASA Open Source*, dan apabila tidak memenuhi target yang telah ditentukan maka proses penelitian akan diulang dari tahap *data preparation* untuk pertama memastikan bahwa data yang digunakan bersih untuk kemudian dilakukan *hyperparameter tuning* kembali pada model. apabila tingkat akurasi pada data validasi sudah memenuhi target maka akan dipastikan kembali apakah terdapat permasalahan dari tujuan bisnis yang belum terselesaikan setelah pemakaian *chatbot* yang lebih masif di kalangan *guest tester*.

2.6. *Deployment*

Dan terakhir pada tahap *deployment* akan dihasilkan sebuah dokumentasi mengenai awal penelitian hingga akhirnya tercipta sebuah *chatbot* yang dapat memenuhi target awal yang sudah ditentukan dalam proses penelitian berdasarkan hasil pengujian akhir. Kemudian untuk menyajikan *chatbot* ini pada publik akan dilakukan proses *hosting* ke *platform cloud service* seperti yang digambarkan pada Gambar 3.



Gambar 3. Arsitektur Sistem *Deployment Chatbot*

Proses *deployment* seperti yang dapat dilihat pada Gambar 3 akan dimulai dari tahap *push project chatbot* RASA X yang telah selesai diuji pada *local computer* ke *repository* GitHub dengan dilengkapi *file docker compose* yang berfungsi untuk mengkonversi *project chatbot* menjadi sebuah *docker*, dimana *docker* sendiri berfungsi sebagai kontainer dari keseluruhan suatu *project* yang akan bertindak seperti sebuah *virtual machine*, sehingga Okteto hanya perlu mengoperasikan *docker*. Dan dari pengaturan *docker compose* dapat diatur agar Okteto bisa memberikan *endpoint API* dari *server chatbot* RASA yang dapat digunakan pada *script widget RASA-webchat* yang telah dipasang pada *website* kampus sebagai tujuan pemasangan *widget chatbot* ini. Dan dengan demikian *chatbot* sudah dapat diakses pada *website* yang telah disiapkan.

3. HASIL DAN PEMBAHASAN

Karna metode *CRISP-DM* yang digunakan pada penelitian ini menggunakan model *RAD*, maka akan dilakukan pengembangan *chatbot* secara berulang hingga iterasi kedua, dimana pada iterasi kedua akan dilakukan pembelajaran pada sampel baru yang didapatkan dari pengguna *chatbot* agar bisa mengembangkan *chatbot* yang lebih bagus dan sesuai dengan pengguna.

3.1. Hasil Pengembangan *Chatbot* di Iterasi Pertama

Pada iterasi pertama, dilakukan tahap *business* dan *data understanding* terlebih dahulu pada sumber informasi yang digunakan dan akan dipilih kategori informasi yang paling penting untuk mahasiswa dan juga ada *intent* percakapan alami untuk menyapa dan sekaligus menjadi batasan apabila pengguna yaitu mahasiswa menanyakan hal yang diluar kemampuan *chatbot*. Dari semua yang telah di diskusikan pada *business* hingga *data understanding* maka didapatkanlah *dataset* awal dengan total 879 sampel kalimat dengan rincian total 38 *intent* yang dibagi menjadi 2 kelompok yaitu kelompok *intent* 'percakapan alami' yang terdiri dari 3 *intent* yaitu 'salam', 'sampai_jumpa', dan 'diluar_informasi' dengan total semuanya 215 sampel kalimat, dan sisa 35 *intent* dengan total 664 sampel kalimat termasuk kedalam kelompok *intent* 'informasi akademik'. Kemudian *dataset* yang telah dibuat akan dipersiapkan untuk dimasukkan pada *folder project chatbot* di tahap *data preparation* ini, dan kemudian akan dipersiapkan *pipeline* dari model *chatbot* agar dapat melakukan *training* model, dimana pada *pipeline* telah dipersiapkan parameter-parameter yang akan di tuning pada proses *hyperparameter tuning* nanti di tahap *evaluation* yaitu *batch size*, *learning rate*, dan *min & max ngram*, serta juga akan digunakan *random seed* untuk memastikan susunan *dataset* pada *training* dan *validation set* tidak berubah untuk menjamin hasil dari *hyperparameter tuning* tidak terpengaruh perubahan susunan data di tiap *set* seperti hasil yang dapat dilihat pada Tabel 1 yang menunjukkan hasil yang memuaskan.

Tabel 1. Hasil Akhir *Hyperparameter Tuning* di Iterasi Pertama Pengembangan *Chatbot*

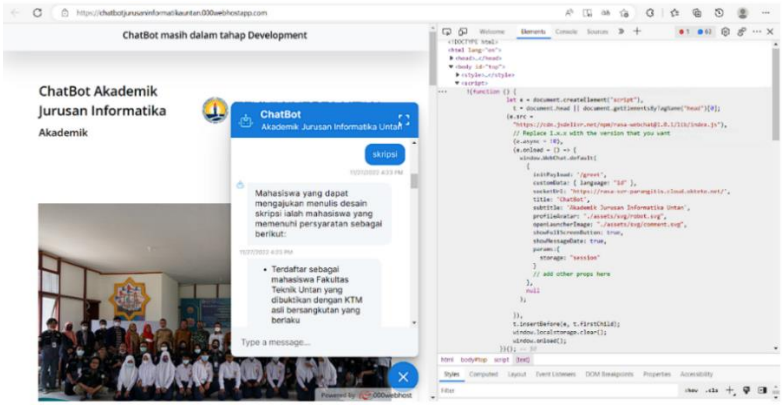
learning rate		0,001
batch size		[150,175]
epochs		171
[min, max] ngram		[1,4]
Hasil dengan *random seed* = 347		
Percobaan ke-	**Akurasi (0-100%)**	**Nilai *loss***
1	98,57%	3,72
Hasil tanpa *random seed*		
Percobaan ke-	**Akurasi (0-100%)**	**Nilai *loss***
1	96,86%	5,577
2	96%	4,751
3	95,14%	6,658
4	98,29%	3,867
5	97,43%	4,296

Berdasarkan pada Tabel 1, dengan berbagai kombinasi *hyperparameter* yang telah digunakan, didapatkan skenario *hyperparameter* terbaik seperti pada Tabel 1 dimana berhasil didapatkan nilai akurasi diatas 95% dan nilai *loss* yang rendah dan selaras dengan tinggi rendahnya nilai akurasi baik dengan *random seed* dan pada 5 percobaan tanpa *random seed*. Namun karna proses *hyperparameter tuning* hanya menggunakan 1 *validation set* untuk diukur tingkat akurasinya, maka diperlukan pengujian dengan *cross validation* yang menggunakan *k-fold* untuk membagi *dataset* menjadi banyak bagian dengan 5 *fold* merupakan jumlah yang paling umum, penggunaan *cross validation* bertujuan agar semua isi *dataset* dapat berperan menjadi sebuah *test set* yang akan dicoba untuk diklasifikasikan oleh *chatbot*. Dari hasil *cross validation* dengan skenario 4, 5, dan 6 *fold* yang dilakukan dengan *dataset* awal dan *pipeline* disini menunjukkan nilai akurasi total yang lebih rendah sekitar 4% dari

DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591

hasil *hyperparameter tuning*, yang masih dapat ditoleransi dikarenakan *cross validation* terhukum lebih besar akibat menggunakan semua kemungkinan sampel kalimat pada *dataset* sebagai *test set* sehingga berisiko terjadinya penurunan akurasi total, namun dengan memanfaatkan hasil analisis *cross validation* yang terperinci di tiap *intent* nya dari *platform RASA Open Source*, dapat ditemukan fakta bahwa tanpa memasukkan kelompok *intent* 'percakapan alami' kedalam perhitungan akurasi total, maka bisa didapatkan peningkatan nilai akurasi sekitar 2%.

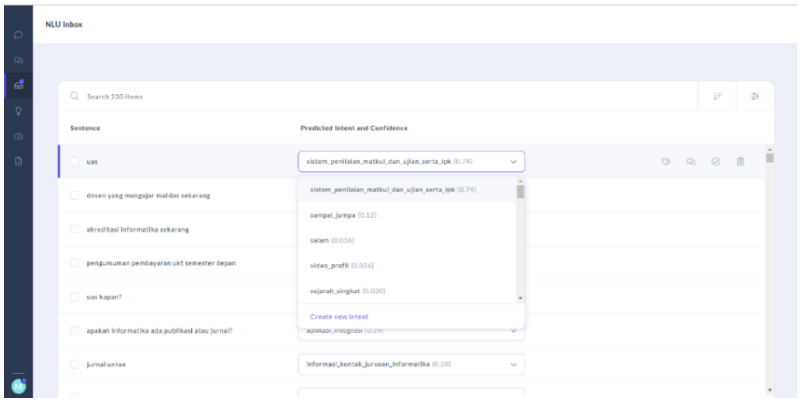
Kemudian masuk ke tahap *deployment* akan dilakukan proses *deploy chatbot* seperti pada Gambar 3 dengan tambahan sebuah *script widget* untuk menampilkan *interface* sebuah *webchat* pada *website* yang dapat diimplementasikan dengan *endpoint* dari *API server chatbot* yang dikembangkan pada *platform RASA Open Source*, dan dalam kasus ini ketika *chatbot* telah di *deploy* ke *platform hosting* Okteto, maka Okteto akan menyediakan *endpoint* khusus yang berisi *API server chatbot RASA Open Source* agar dapat terus berjalan secara daring dan dapat dipasang pada *script* tersebut hingga dapat ditampilkan seperti pada Gambar 4.



Gambar 4. Tampilan *Widget* berupa *Webchat RASA Open Source* pada sebuah *Website* beserta *Script Widget RASA-webchat*

Dapat terlihat tampilan *website* ujicoba *chatbot* beserta *widget chatbot* yang telah berfungsi, dan tampilan *script widget RASA-webchat* yang menyediakan fitur-fitur kostumisasi tampilan *widget chatbot* dan variabel 'socketUrl' yang diisi oleh *API server chatbot RASA Open Source* yang didapatkan dari Okteto.

Setelah *chatbot* berhasil di *deploy*, akan dilakukan pengujian *chatbot* oleh pengguna yaitu mahasiswa Jurusan Informatika Fakultas Teknik Universitas Tanjungpura. Pengujian yang dilakukan dapat dipantau hasilnya dengan memanfaatkan *interface GUI* pengembangan *chatbot* yang dimiliki *RASA Open Source*, yaitu RASA X dimana dapat dilakukan pemantauan sampel kalimat baru apa saja yang diterima oleh *chatbot* dan dapat dijadikan bahan untuk dipelajari oleh *chatbot* di pengembangan lanjutan seperti pada Gambar 5.



Gambar 5. *NLU Inbox* pada RASA X

DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591

Pada Gambar 5, dapat diketahui bahwa fitur *NLU Inbox* pada RASA X dapat menampung segala kalimat yang pernah dimasukkan oleh pengguna ke *chatbot* dan dapat diketahui sebaran nilai *confidence* di tiap *intent* nya.

Pengujian pertama *chatbot* ke pengguna berhasil mendapatkan sampel kalimat baru sebanyak 227 sampel dengan 30 sampel diantaranya tidak memiliki arti yang jelas atau ambigu sehingga tidak dapat dimasukkan kedalam pengujian klasifikasi *intent* sampel baru, begitu juga dengan 29 sampel lain yang teridentifikasi sebagai

Pengujian pertama *chatbot* ke pengguna berhasil mendapatkan sampel kalimat baru sebanyak 227 sampel dengan 30 sampel diantaranya tidak memiliki arti yang jelas atau ambigu sehingga tidak dapat dimasukkan kedalam pengujian klasifikasi *intent* sampel baru, begitu juga dengan 29 sampel lain yang teridentifikasi sebagai *intent* yang tidak pernah dipelajari *chatbot* yang akan dijadikan bahan pembelajaran untuk membentuk *intent* baru untuk *dataset* terbaru nanti. Sehingga pengujian pertama *chatbot* untuk mengklasifikasi *intent* dari sampel baru akan menyisakan 168 sampel untuk coba diklasifikasi, dengan rincian hasilnya dapat dilihat pada Tabel 2.

Tabel 2. Hasil Pengujian Klasifikasi *Intent* yang Pertama

Skenario	**Akurasi Total**	***Macro-average***			***Weighted-average***		
		precision	***recall***	***f1-score***	***precision***	***recall***	***f1-score***
Seluruh *intent*	72,02%	75,45%	79,78%	72,50%	81,95%	72,02%	72,48%
Tanpa *intent* 'percakapan alami'	76,39%	79,29%	82,12%	76,02%	88,53%	76,39%	78,83%

Pada Tabel 2, nilai akurasi total dari skenario tanpa *intent* 'percakapan alami' lebih tinggi dari skenario penggunaan seluruh *intent*, begitu juga pada *macro* dan *weighted average*, ini menunjukkan bahwa *intent* 'percakapan alami' cukup sulit untuk diklasifikasikan dengan benar oleh *chatbot* yang dapat dikarenakan *feature* dari tiap kalimat pada *intent* tersebut yang sangat sedikit untuk dipelajari *chatbot*.

Setelah pengujian klasifikasi sampel pengguna yang pertama selesai, akan didapatkan *confusion matrix* dan *histogram intent* yang akan ditampilkan nanti sebagai perbandingan dengan yang didapatkan dari pengujian klasifikasi sampel pengguna yang kedua.

3.2. Hasil Pengembangan *Chatbot* di Iterasi Kedua

Pengembangan sekarang masuk ke iterasi kedua, dimana seluruh sampel baru dari pengguna pada tahap *business* dan *data understanding* di iterasi kedua dan seterusnya akan dikembangkan untuk digabungkan ke *dataset* awal, didapatkan *dataset* terbaru dengan total 1210 sampel kalimat (bertambah 331 sampel) dengan rincian total 40 *intent* yang dibagi menjadi 2 kelompok yaitu kelompok *intent* 'percakapan alami' yang tetap terdiri dari 3 *intent* yang sama dengan total semuanya 237 sampel kalimat (bertambah 22 sampel), dan sisa 37 *intent* dengan total 973 *intent* (bertambah 309 sampel) sampel kalimat termasuk kedalam kelompok *intent* 'informasi akademik' yang mendapatkan 2 *intent* baru beserta 2 *intent* lama yang digabungkan dengan sub *intent* baru yang cocok untuk digabung dengan beberapa *intent* lama. Kemudian dilakukan *data preparation* seperti pada iterasi pertama dan *pipeline* yang sudah diatur dari iterasi pertama, maka dapat langsung dilakukan *hyperparameter tuning* dengan hasil yang dapat dilihat pada Tabel 3.

Apabila dilakukan perbandingan dengan hasil *hyperparameter tuning* di iterasi pertama pada Tabel 1, maka hasil *hyperparameter tuning* di iterasi kedua pada Tabel 3 baik di nilai akurasi dan nilai *loss* dengan menggunakan *random seed* dan tanpa *random seed* sedikit menurun. Ini dapat disebabkan oleh lebih bervariasinya kalimat baru yang didapatkan dari iterasi pertama dimana *dataset* menjadi lebih natural berkat tambahan kalimat dari pengguna dimana fitur dari tiap kalimat dalam satu kelas *intent* menjadi lebih berbeda satu sama lainnya dibandingkan apabila *dataset* dibuat oleh satu orang saja seperti yang dibuat pada iterasi pertama.

DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591

Tabel 3. Hasil Akhir *Hyperparameter Tuning* di Iterasi Kedua Pengembangan *Chatbot*

learning rate	0,003
batch size	[206,228]
epochs	133
[min, max] ngram	[1,4]

Hasil dengan *random seed* = 347		
Percobaan ke-	**Akurasi (0-100%)**	**Nilai *loss***
1	97,37%	4,427
Hasil tanpa *random seed*		
Percobaan ke-	**Akurasi (0-100%)**	**Nilai *loss***
1	96,49%	8,15
2	94,445%	5,951
3	92,7%	8,239
4	92,04%	7,745
5	96,05%	7,45

Kemudian dari hasil *cross validation* dengan skenario 4, 5, dan 6 *fold* yang dilakukan dengan *dataset* terbaru dan *pipeline* yang telah dilakukan *hyperparameter tuning* lagi tidak menunjukkan perbedaan yang signifikan dengan hasil yang didapat di iterasi pertama meskipun terdapat penurunan sekitar 1-2% namun tetap masih memiliki karakteristik yang sama, dan fakta mengenai kelompok *intent* 'percakapan alami' memperburuk nilai akurasi total masih terbukti pada iterasi kedua ini dimana tanpa menggunakan kelompok *intent* 'percakapan alami' nilai akurasi total meningkat sekitar 2%.

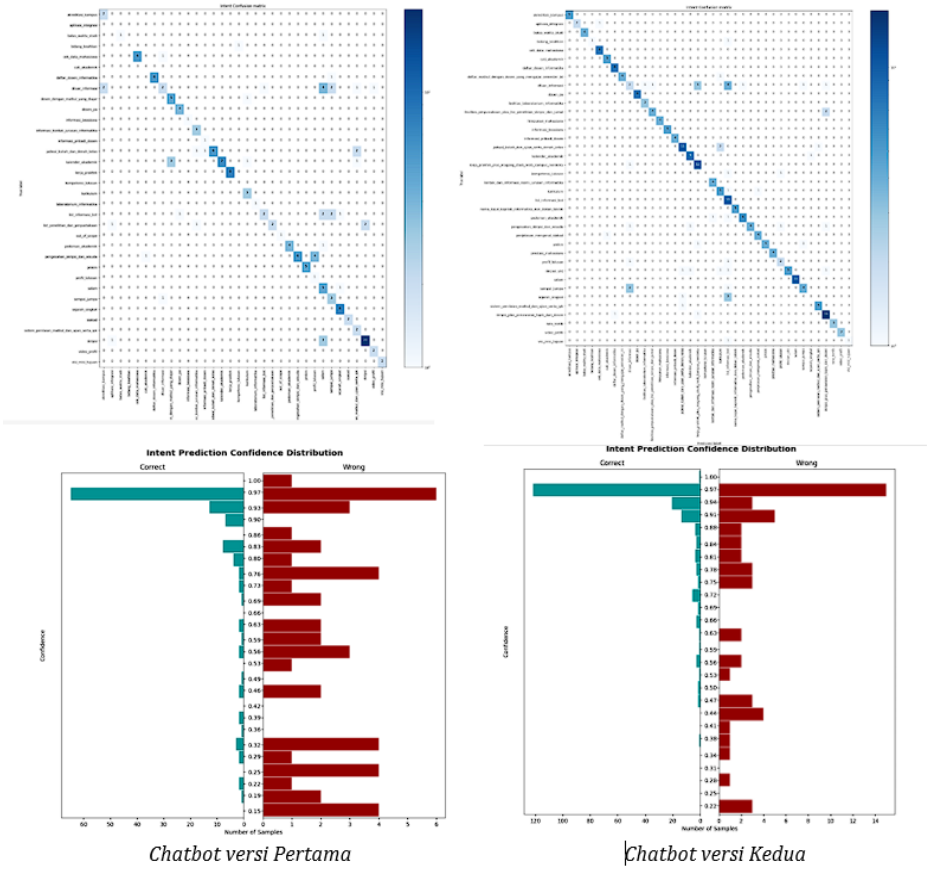
Chatbot dengan model terbaru hasil *hyperparameter tuning* dan *cross validation* di iterasi kedua akan di *deploy* untuk menggantikan *chatbot* versi pertama. *Chatbot* terkini yang dapat dianggap sebagai *chatbot* versi ke 2 ini akan dilakukan pengujian ke pengguna seperti pada *chatbot* versi pertama, dimana pada pengujian *chatbot* terbaru ini ke pengguna berhasil mendapatkan sampel kalimat baru sebanyak 302 sampel (75 sampel lebih banyak dari pengujian *chatbot* pertama) dengan 11 sampel (19 sampel lebih sedikit dari pengujian *chatbot* pertama) diantaranya tidak memiliki arti yang jelas atau ambigu sehingga tidak dapat dimasukkan kedalam pengujian klasifikasi *intent* sampel baru, begitu juga dengan 41 sampel lain yang teridentifikasi sebagai *intent* yang tidak pernah dipelajari *chatbot* (12 sampel lebih banyak dari *chatbot* pertama) yang akan dijadikan bahan pembelajaran untuk membentuk *intent* baru untuk *dataset* terbaru lagi nanti. Sehingga pengujian kedua *chatbot* untuk mengklasifikasi *intent* dari sampel baru akan menyisakan 250 sampel untuk coba diklasifikasi, dengan rincian hasilnya dapat dilihat pada Tabel 4.

Tabel 4. Hasil Pengujian Klasifikasi *Intent* yang Kedua

Skenario	**Akurasi Total**	***Macro-average***			***Weighted-average***		
		precision	***recall***	***f1-score***	***precision***	***recall***	***f1-score***
Seluruh *intent*	78,40%	81,36%	77,73%	77,87%	79,18%	78,40%	77,03%
Tanpa *intent* 'percakapan alami'	82,49%	82,86%	79,70%	79,53%	82,08%	82,49%	80,40%

Apabila dilakukan perbandingan hasil pengujian klasifikasi *intent* dari *chatbot* versi pertama pada Tabel 2 dan kedua yang ada pada Tabel 4 diatas, maka didapatkan fakta bahwa hasil pengujian klasifikasi *intent* pada *chatbot* terbaru dengan 82 sampel lebih banyak yang diujikan, berhasil mendapatkan nilai akurasi total yang 6% lebih bagus daripada *chatbot* pertama di setiap skenario, dan terbukti juga bahwa di kedua versi *chatbot*, skenario tanpa penggunaan kelompok *intent* 'percakapan alami' dapat meningkatkan nilai akurasi total sekitar 4%, begitu juga dengan *macro* dan *weighted average* yang rata-rata lebih bagus pada *chatbot* versi kedua, sehingga ini membuktikan bahwa ukuran *dataset* sangat berpengaruh untuk bisa mendapatkan performa *chatbot* yang diinginkan. Dan juga dapat dilakukan perbandingan hasil *confusion matrix* dan *histogram intent* dari hasil pengujian klasifikasi *intent* dari kedua versi *chatbot* tersebut.

DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591



Gambar 6. Perbandingan *Confusion Matrix* (atas) dan *Histogram Intent* (bawah) dari Hasil Pengujian Klasifikasi *Intent* Sampel Baru oleh *Chatbot* Awal dan Terbaru

Dapat dilihat di Gambar 6, dimana pada *confusion matrix* apabila kotak-kotak yang membentuk garis diagonalnya lebih tebal dibanding *confusion matrix* yang lain, itu menandakan *chatbot* dengan *confusion matrix* yang garis diagonalnya lebih tebal dan tidak terdapat banyak titik baik itu yang tipis dan tebal diluar garis diagonal memiliki performa yang lebih bagus, dan terlihat bahwa pada *chatbot* versi kedua(terbaru) memiliki garis diagonal yang lebih tebal dan tidak banyak titik diluar garis diagonal dibanding *chatbot* versi pertama. Dan pada *histogram intent* dapat dilihat persebaran klasifikasi *intent* yang benar dan salah beserta rentang nilai *confidence* nya yang kurang adil untuk dibandingkan karena jumlah sampel yang coba diklasifikasi oleh kedua versi *chatbot* ini cukup berbeda jauh, sehingga biasanya *histogram intent* lebih digunakan sebagai alat bantu untuk menentukan *threshold* dari nilai *confidence* klasifikasi suatu *intent* yang dapat memicu *Two stage fallback*, sehingga ketika suatu klasifikasi *intent* terhadap sebuah sampel memiliki nilai *confidence* di bawah *threshold* yang sudah ditentukan, maka *Two stage fallback* akan aktif.

3.3. Pembahasan

Pengembangan *chatbot* pertama berhasil mendapatkan hasil yang bagus pada *hyperparameter tuning* dan *cross validation*, dan dengan bermodalkan sampel kalimat baru dari pengguna *chatbot* pertama, maka dapat dilakukan pengembangan lanjutan untuk menghasilkan *chatbot* versi terbaru dengan *dataset* yang lebih besar dan bervariasi dalam hal *intent* yang lebih banyak dan karakteristik sampel baru yang tidak ada di *dataset* awal [13].

Dari hasil *hyperparameter tuning* yang dilakukan dalam pengembangan *chatbot* lanjutan, dapat diambil kesimpulan bahwa nilai *hyperparameter* perlu diperbesar apabila ukuran *dataset* lebih besar dari sebelumnya, namun terkhusus *hyperparameter batch size* akan diperlukan penyesuaian kedepannya apabila mesin komputer yang digunakan untuk *training chatbot* sudah tidak mampu untuk

DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591

menggunakan *batch size* yang lebih besar dari sebelumnya, sehingga kedepannya perlu dilakukan penelitian untuk menemukan nilai *batch size* kecil untuk menangani *validation set* yang lebih besar lagi kedepannya, dikarenakan *batch size* paling optimal sejauh ini adalah *batch size* yang nilainya mendekati ukuran *validation set*, dan nilai *learning rate* akan mengikuti naik turunnya nilai *batch size* sesuai dengan hasil penelitian yang didapatkan [14].

Dari hasil *cross validation* yang didapatkan pada pengembangan *chatbot* lanjutan juga menunjukkan bahwa hasil hyperparamter tuning sudah cukup memuaskan dikarenakan tidak berbeda jauh dengan hasil *cross validation* pada pengembangan awal *chatbot*, ini berarti *hyperparameter tuning* yang dilakukan pada pengembangan lanjutan *chatbot* sudah menyesuaikan dengan *dataset* baru hasil penggabungan *dataset* awal dengan kumpulan sampel baru hasil pembelajaran sampel kalimat dari pengguna *chatbot* awal.

Masa-masa awal pengembangan *chatbot* akan sangat cepat, ini dikarenakan masih sedikitnya pengetahuan yang dipelajari *chatbot* dalam bentuk sebuah *dataset* yang dapat dikembangkan seiring waktu dengan memanfaatkan sumber pengetahuan yang berupa *NLU Inbox* dari RASA X yang akan menampung segala sampel kalimat baru yang belum pernah dipelajari sebelumnya yang dikirim oleh pengguna [15], sehingga dapat dipastikan di awal pengujian *chatbot* kepada pengguna banyak, akan masuk sampel kalimat baru dalam jumlah yang banyak sehingga proses pengembangan *chatbot* akan sangat cepat dan masif di beberapa bulan awal pengembangan. Namun setelah itu proses pengembangan *chatbot* cenderung akan mulai melambat dikarenakan sudah hampir semua sampel yang berhubungan dengan layanan akademik Jurusan Informatika Universitas Tanjungpura dipelajari oleh *chatbot*, sehingga dengan adanya juga batasan informasi yang dapat dipelajari oleh *chatbot* yang hanya berkisar di area lingkup akademik Jurusan, maka pengembangan *chatbot* secara masif kedepannya tidak akan begitu lama.

4. KESIMPULAN

Berdasarkan pengembangan yang telah dilakukan, penelitian awal dimulai dari tahap *business* dan *data understanding* dimana dilakukan pengembangan *dataset* awal yang bersumber dari buku pedoman akademik dan *website* Jurusan Informatika Universitas Tanjungpura hingga diperoleh 879 sampel kalimat yang terdiri dari 38 *intent*, dan pada iterasi kedua berkat hasil pengujian *chatbot* yang telah di *deploy* agar dapat diuji oleh pengguna di iterasi pertama, dihasilkan *dataset* terbaru yang lebih besar dan bervariatif dengan memiliki 1210 sampel kalimat(bertambah 331 sampel) yang terdiri dari 40 *intent*. Dan dengan kedua *dataset* tersebut, dilakukan *hyperparameter tuning* dan *cross validation* pada *dataset* awal di iterasi pertama dan *dataset* terbaru di iterasi kedua. Kedua iterasi berhasil mendapatkan rata-rata akurasi di atas 90% pada *hyperparameter tuning* dan *cross validation* yang dapat dicapai dengan cara menyesuaikan ukuran *batch size* dan *learning rate* dengan semakin besarnya ukuran *dataset* di tiap iterasi berikutnya. Dan ketika *chatbot* diujikan dalam mengklasifikasi sampel kalimat dari pengguna, nilai akurasi total akan mengalami penurunan yang cukup drastis dibandingkan dengan hasil *hyperparameter tuning* dan *cross validation* dikarenakan sampel kalimat dari pengguna langsung akan lebih bervariatif dan akan lebih banyak kalimat dan *intent* baru yang belum pernah dipelajari *chatbot*. Namun dengan seiring bertambahnya iterasi pengembangan, akurasi pun akan semakin meningkat drastis terutama di beberapa iterasi awal, yang terbukti dengan didapatkannya peningkatan akurasi total sebesar 6% dari iterasi pertama ke iterasi kedua.

DAFTAR PUSTAKA

[1] A. Massaro, V. Maritati, and A. Galiano, “Automated Self-learning Chatbot Initially Build as a FAQs Database Information Retrieval System: Multi-level and Intelligent Universal Virtual Front-office Implementing Neural Network,” *Informatica*, vol. 42, no. 4, Nov. 2018, doi: 10.31449/inf.v42i4.2173.

[2] J. A. Mulyono and S. Sfenrianto, “Evaluation of Customer Satisfaction on Indonesian Banking Chatbot Services During the COVID-19 Pandemic,” *CommIT (Communication Inf. Technol. J.*, vol. 16, no. 1, pp. 69–85, Mar. 2022, doi: 10.21512/commit.v16i1.7813.

DOI: https://doi.org/10.54082/jiki.73
P-ISSN: 2807-6664
E-ISSN: 2807-6591

[3] S. Yang and K. Stansfield, "AI Chatbot for Educational Service Improvement in the Post-Pandemic Era: A Case Study Prototype for Supporting Digital Reading List," in *2022 13th International Conference on E-Education, E-Business, E-Management, and E-Learning (IC4E)*, Jan. 2022, pp. 24–29. doi: 10.1145/3514262.3514289.

[4] A. Abdellatif, K. Badran, D. E. Costa, and E. Shihab, "A Comparison of Natural Language Understanding Platforms for Chatbots in Software Engineering," *IEEE Trans. Softw. Eng.*, vol. 48, no. 8, pp. 3087–3102, Dec. 2022, doi: 10.1109/TSE.2021.3078384.

[5] B. R. Ranoliya, N. Raghuwanshi, and S. Singh, "Chatbot for university related FAQs," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2017, pp. 1525–1530. doi: 10.1109/ICACCI.2017.8126057.

[6] A. Jiao, "An Intelligent Chatbot System Based on Entity Extraction Using RASA NLU and Neural Network," *J. Phys. Conf. Ser.*, vol. 1487, no. 1, p. 012014, Mar. 2020, doi: 10.1088/1742-6596/1487/1/012014.

[7] B. Luo, R. Y. K. Lau, C. Li, and Y. Si, "A critical review of state-of-the-art chatbot designs and applications," *WIREs Data Min. Knowl. Discov.*, vol. 12, no. 1, Jan. 2022, doi: 10.1002/widm.1434.

[8] T. Wochinger, "Failing Gracefully with Rasa," *Rasa Blog*, 2019. https://medium.com/rasa-blog/failing-gracefully-with-rasa-8ead6b43f2f4 (accessed Jan. 25, 2019).

[9] Y. Sumikawa, M. Fujiyoshi, H. Hatakeyama, and M. Nagai, "Supporting Creation of FAQ *Dataset* for E-Learning Chatbot," *Intelligent Decision Technologies 2019*, pp. 3–13, 2020. doi: 10.1007/978-981-13-8311-3_1.

[10] C. Greyling, "Rasa-X Is A Unique Approach To Continuous Chatbot Improvement," *Medium*, 2020. https://cobusgreyling.medium.com/rasa-x-has-a-unique-approach-to-continuous-chatbot-improvement-420a367f4146 (accessed Aug. 14, 2020).

[11] A. Weidauer, "Conversational AI: Your Guide to Five Levels of AI Assistants in Enterprise," *Rasa Blog*, 2018. https://rasa.com/blog/conversational-ai-your-guide-to-five-levels-of-ai-assistants-in-enterprise/ (accessed Sep. 27, 2018).

[12] S. Studer *et al.*, "Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology," *Mach. Learn. Knowl. Extr.*, vol. 3, no. 2, pp. 392–413, Apr. 2021, doi: 10.3390/make3020020.

[13] Z. H. Pradana, H. Nafi'ah, and R. A. Rochmanto, "in Chatbot-based Information Service using RASA Open-SourceFrameworkin Prambanan Temple Tourism Object," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 4, pp. 656–662, Aug. 2022, doi: 10.29207/resti.v6i4.3913.

[14] Y. Windiatmoko, R. Rahmadi, and A. F. Hidayatullah, "Developing Facebook Chatbot Based on Deep Learning Using RASA Framework for University Enquiries," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1077, no. 1, p. 012060, Feb. 2021, doi: 10.1088/1757-899X/1077/1/012060.

[15] D. G. S. Ruindungan and A. Jacobus, "Chatbot Development for an Interactive Academic Information Services using the Rasa Open Source Framework," *J. Tek. Elektro dan Komput.*, vol. 10, no. 1, pp. 61–68, 2021.