

PERBANDINGAN ALGORITMA RATCLIFF/OBERSHELP DAN LEVENSHEIN DISTANCE DALAM CHATBOT AIML UNTUK UNIVERSITAS XYZ

Pandu Zidni¹, Agus Iskandar^{2*}

^{1,2}Universitas Nasional, Indonesia

¹panduzidni02@gmail.com, ²agus.iskandar@civitas.unas.ac.id

Received: 17-02-2025

Revised: 25-02-2025

Approved: 02-03-2025

ABSTRAK

Pelayanan informasi akademik yang efektif dan responsif merupakan kebutuhan penting bagi institusi pendidikan tinggi untuk mendukung kebutuhan informasi mahasiswa. Namun, sistem berbasis web yang tersedia di Universitas XYZ masih memiliki keterbatasan karena sifatnya yang satu arah dan kurang fleksibel dalam memberikan respons langsung terhadap pertanyaan, terutama di luar jam operasional. Penelitian ini mengembangkan chatbot berbasis Artificial Intelligence Markup Language (AIML) sebagai solusi pelayanan informasi akademik dua arah. Fokus penelitian ini adalah membandingkan dua algoritma dari konsep text similarity, yaitu Ratcliff/Obershelp dan Levenshtein Distance, untuk meningkatkan akurasi dan efisiensi chatbot dalam memahami dan merespons pertanyaan mahasiswa. Pengembangan chatbot ini menggunakan pustaka Sastrawi untuk proses stemming Bahasa Indonesia, serta JSON untuk penanganan sinonim. Algoritma Ratcliff/Obershelp dan Levenshtein Distance diterapkan untuk meningkatkan akurasi jawaban chatbot dalam mengenali berbagai variasi kata pada pertanyaan pengguna. Kedua algoritma ini digunakan untuk menghitung tingkat kemiripan antara input pengguna dan pola yang ada, sehingga chatbot mampu memberikan respons yang relevan dan sesuai dengan kebutuhan mahasiswa Universitas XYZ. Penggunaan chatbot berbasis AIML dengan algoritma yang dibandingkan ini diharapkan dapat menjadi referensi bagi pengembangan sistem pelayanan informasi yang lebih interaktif dan efisien di lingkungan akademik.

Kata Kunci: Chatbot, AIML, Natural Language Processing, Ratcliff/Obershelp, Levenshtein Distance

PENDAHULUAN

Banyak universitas mengandalkan sistem informasi web dan media sosial untuk menyampaikan informasi akademik, pendaftaran, beasiswa, dan biaya pendidikan [1].

Calon mahasiswa membutuhkan komunikasi dua arah untuk mendapatkan informasi lebih mendalam [2]. Situs web kampus sebagai sumber utama sering kali bersifat statis dan satu arah, sehingga tidak selalu menjawab pertanyaan spesifik. Akibatnya, mahasiswa harus menghubungi administrasi kampus, yang bisa memakan waktu lama, terutama di luar jam kerja atau saat hari libur.

Chatbot adalah sistem berbasis kecerdasan buatan dan pemrosesan bahasa alami yang memungkinkan interaksi manusia dan mesin [3]. Chatbot semakin populer dalam layanan pelanggan, pencarian informasi, dan interaksi personal. Keunggulannya meliputi ketersediaan 24/7 serta kemampuan menangani banyak pertanyaan sekaligus, memungkinkan sumber daya manusia fokus pada tugas yang lebih kompleks [4].

Chatbot, sebagai sistem percakapan virtual non-manusia, kini mendapat perhatian luas di bidang akademis dan praktis karena kemampuannya menjawab pertanyaan pengguna [5].

Teknologi chatbot yang berbasis Artificial Intelligence Markup Language (AIML) muncul sebagai solusi untuk menyediakan layanan informasi dua arah [6]. Dengan AIML, chatbot dapat merespons pertanyaan mahasiswa secara dinamis menggunakan pendekatan pattern matching [7], yaitu dengan mencocokkan pertanyaan yang diajukan dengan pola-pola tertentu untuk menghasilkan jawaban yang relevan [8].

Chatbot merupakan bagian dari Natural Language Processing (NLP), yang merupakan suatu sistem yang mampu menganalisis teks dengan menggunakan berbagai teori dan teknik [9]. NLP mencakup berbagai bidang, seperti pengambilan informasi, ekstraksi informasi, serta sistem tanya jawab [10][11].

Penggunaan Natural Language Processing (NLP) penting untuk mengembangkan chatbot cerdas yang memahami variasi bahasa manusia [12]. Dalam bahasa Indonesia, pustaka Sastrawi berperan sebagai stemmer untuk mengubah kata berawalan menjadi bentuk dasarnya [13][14].

Penelitian sebelumnya menunjukkan keterbatasan chatbot AIML dalam akurasi dan fleksibilitas memahami pertanyaan. Penelitian yang dilakukan oleh [11] mengembangkan chatbot AIML dengan Named Entity Recognition (NER) berbasis Hidden Markov Model (HMM) untuk menjawab pertanyaan akademik di UIN Suska Riau, menggunakan 150 kategori pertanyaan dan 30 jawaban dari panduan akademik.

Penelitian sebelumnya memiliki keterbatasan, dengan akurasi chatbot hanya 55%, menyebabkan banyak pertanyaan tidak terjawab. Chatbot hanya merespons jika pola cocok dengan kategori AIML, sehingga kurang fleksibel dalam menangani variasi bahasa. Selain itu, chatbot sebelumnya tidak menggunakan text similarity, sehingga sulit mengenali pertanyaan serupa dengan struktur berbeda.

Penelitian ini mengusulkan pendekatan baru dengan algoritma text similarity Ratcliff/Obershelp [15] dan Levenshtein Distance [16] untuk meningkatkan akurasi chatbot AIML. Metode ini membantu chatbot memahami pertanyaan meski ada variasi struktur atau kata [17]. Selain itu, integrasi stemming dan sinonim memperkuat pengenalan berbagai bentuk kata dalam Bahasa Indonesia [18].

Pendekatan ini bertujuan meningkatkan akurasi dan efisiensi chatbot AIML dalam menjawab berbagai variasi pertanyaan mahasiswa. Chatbot yang dikembangkan diharapkan lebih responsif dalam memenuhi kebutuhan informasi akademik di Universitas XYZ.

METODE PENELITIAN

Penelitian ini menggunakan kombinasi metode Research and Development (R&D) [19] dan System Development Life Cycle (SDLC) [20]. R&D digunakan untuk eksplorasi teknologi chatbot AIML, memahami kebutuhan pengguna, dan mengumpulkan data. Selanjutnya, SDLC diterapkan dalam pengembangan chatbot secara bertahap, mulai dari perencanaan hingga pengujian, guna memastikan kualitas dan kesesuaian dengan kebutuhan pengguna [21]. Tahapan penelitian ini dijelaskan lebih lanjut pada Gambar 1.



Gambar 1 Metode Penelitian

Penelitian dimulai dengan persiapan pengumpulan data, sumber daya, dan perencanaan alur penelitian. Masalah utama yang diidentifikasi adalah kebutuhan mahasiswa akan akses cepat dan akurat terhadap informasi akademik, di mana chatbot AIML sebelumnya memiliki akurasi hanya 55%. Penelitian bertujuan meningkatkan pemahaman chatbot melalui penerapan algoritma Ratcliff/Obershelp dan Levenshtein Distance. Peneliti menganalisis referensi terkait chatbot, AIML, NLP, dan algoritma text similarity untuk menentukan metode terbaik dalam pengembangan sistem.

Data dikumpulkan dari pertanyaan umum mahasiswa tentang layanan akademik, termasuk variasi frasa dan sinonim [22]. Desain sistem mencakup arsitektur chatbot, flowchart percakapan, serta implementasi AIML, text similarity, Ratcliff/Obershelp, dan Levenshtein Distance. Stemming dan sinonim diterapkan untuk meningkatkan akurasi. Pengujian fungsional menilai akurasi, responsivitas, dan perbandingan kedua algoritma.

Hasil pengujian dianalisis berdasarkan akurasi, kecepatan respons, dan kemampuan mengenali variasi pertanyaan. Kesimpulan penelitian merangkum pencapaian, tantangan, dan efektivitas algoritma dalam meningkatkan akurasi chatbot Universitas XYZ. Laporan penelitian disusun untuk menyajikan hasil yang dapat digunakan sebagai solusi chatbot informasi mahasiswa.

HASIL DAN PEMBAHASAN

Analisa Pertanyaan Berbeda Gaya Penulisan Pada Algoritma Ratcliff/Obershelp dan Levenshtein Distance



Gambar 2 Analisa Beda Gaya Penulisan

Tahap pertama, Pada gambar 2 pengujian kedua algoritma ini menunjukkan bahwa chatbot dapat mengenali pertanyaan dengan struktur berbeda, seperti "Bagaimana cara mendaftar menjadi mahasiswa?" dan "Saya ingin daftar jadi mahasiswa Informatika di Universitas XYZ, bagaimana caranya?", dan memberikan jawaban yang sesuai. Ini menandakan bahwa kedua algoritma chatbot dapat menangani variasi susunan kata dan struktur kalimat pengguna dengan baik.

Analisa Pertanyaan Kesalahan Ketik Pada Algoritma Ratcliff/Obershelp dan Levenshtein Distance



Gambar 3 Analisa Kesalahan Ketik

Kedua algoritma, Ratcliff/Obershelp dan Levenshtein Distance dapat menangani kesalahan ketik. Ratcliff/Obershelp lebih tahan terhadap variasi kesalahan dengan tingkat kesamaan 65%, sedangkan Levenshtein Distance lebih sensitif terhadap perubahan karakter dengan tingkat kemiripan 57%.

Analisa Kinerja Sinonim Pada Algoritma Ratcliff/Obershelp dan Levenshtein Distance



Gambar 4 Analisa Kinerja Sinonim

Kedua algoritma, Ratcliff/Obershelp dan Levenshtein Distance dapat mengenali sinonim dan frasa serupa. Ratcliff/Obershelp menunjukkan fleksibilitas dalam memahami variasi bahasa, sementara Levenshtein Distance juga mampu memberikan respons yang relevan.

Analisa Kinerja Kata Pemisah Pada Algoritma Ratcliff/Obershelp dan Levenshtein Distance



Gambar 5 Analisa Kinerja Kata Pemisah

Kedua algoritma, Ratcliff/Obershelp dan Levenshtein Distance dapat menangani multiple pertanyaan dalam satu input. Ratcliff/Obershelp memberikan respons berurutan dengan jelas, sementara Levenshtein Distance juga mampu memisahkan pertanyaan meskipun lebih fokus pada pencocokan karakter.

Analisa Kinerja Deteksi Konteks Pada Algoritma Ratcliff/Obershelp dan Levenshtein Distance

Ratcliff/Obershelp



Gambar 6 Analisa Kinerja Deteksi Konteks Ratcliff/Obershelp

Levenshtein Distance



Gambar 7 Analisa Kinerja Deteksi Konteks Levenshtein Distance

Kedua algoritma, Ratcliff/Obershelp dan Levenshtein Distance, menunjukkan efektivitas dalam mendeteksi konteks pertanyaan dan memberikan respons yang relevan. Dalam pengujian, keduanya berhasil mengidentifikasi kata kunci seperti "syarat pendaftaran" dan mempersempit pencocokan pola AIML untuk meningkatkan akurasi respons.

Analisa Keluaran Algoritma Ratcliff/Obershelp dan Levenshtein Distance

Disini melakukan analisa dengan tiga cara yaitu pertanyaan normal, pertanyaan salah ketik, dan dua pertanyaan dalam satu input. Untuk melakukan analisa kita menggunakan rumus menghitung rata-rata untuk menghitung rata-rata dan rumus standar deviasi.

$$\mu = \frac{\sum X_i}{n}$$

Rumus Mean

$$\sigma^2 = \sqrt{\frac{\sum (X_i - \mu)^2}{n - 1}}$$

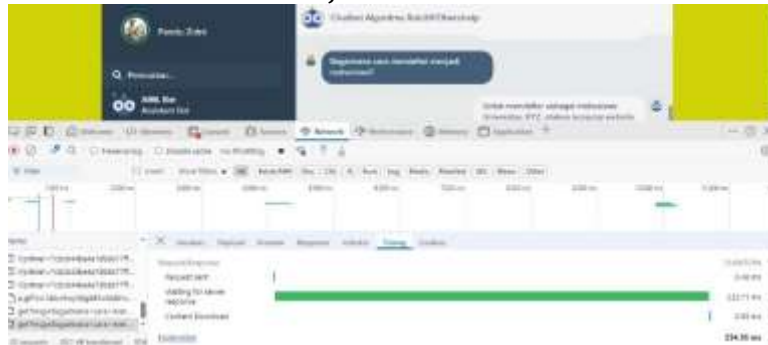
Rumus Standar Deviasi

Uji pertama pada algoritma Ratcliff/Obershelp dengan Levenshtein Distance pada pertanyaan normal:

Uji Pertama Ratclif/Obershelp dengan Pertanyaan Normal

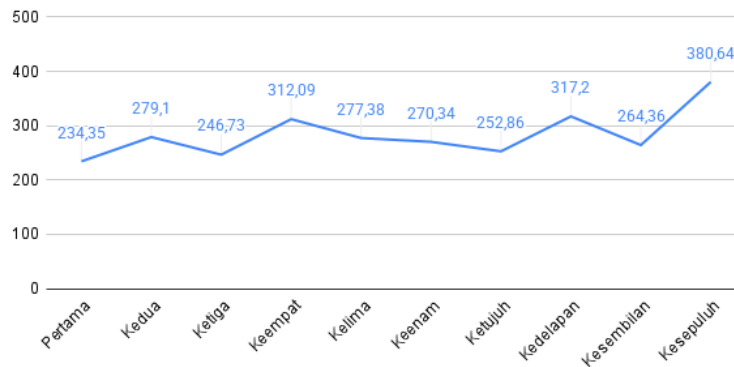
Uji selanjutnya pada algoritma Ratclif/Obershelp pada pertanyaan normal:

Input: “Bagaimana cara mendaftar menjadi mahasiswa?”



Gambar 8 Tangkapan Layar Kecepatan Menjawab Ratcliff/Obershelp

Waktu Respons Uji Pertama Ratcliff/Obershelp dengan Pertanyaan Normal



Gambar 9 Waktu Menjawab Uji Pertama Ratcliff/Obershelp

waktu tercepat yang diperoleh adalah 234,35 ms, sedangkan waktu terlama adalah 380,64 ms, Selanjutnya hitung rata-ratanya:

$$\mu = \frac{234,35 + 279,1 + 246,73 + 312,09 + \dots + 380,64}{10} = 283,50 \text{ ms}$$

Lalu hitung standar deviasi

Tabel 1 Perhitungan Standar Deviasi Ratcliff/Obershelp Pertama

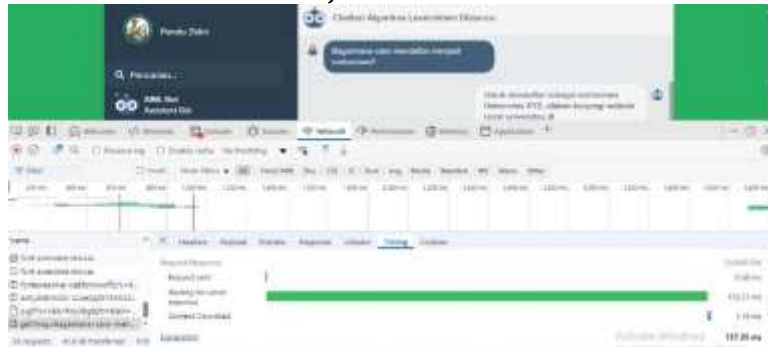
Percobaan	Waktu Respons (ms)	$X_i - \mu$	$(X_i - \mu)^2$
Pertama	234,35	$234,35 - 283,50 = -49,16$	$(-49,16)^2 = 2415,71$
Kedua	279,1	$279,10 - 283,50 = -4,40$	$(-4,40)^2 = 19,40$
Ketiga	246,73	$246,73 - 283,50 = -36,78$	$(-36,78)^2 = 1352,40$
Keempat	312,09	$312,09 - 283,50 = 28,59$	$(28,59)^2 = 817,10$
Kelima	277,38	$277,38 - 283,50 = -6,12$	$(-6,12)^2 = 37,52$
Keenam	270,34	$270,34 - 283,50 = -13,17$	$(-13,17)^2 = 173,32$
Ketujuh	252,86	$252,86 - 283,50 = -30,64$	$(-30,64)^2 = 939,12$
Kedelapan	317,2	$317,20 - 283,50 = 33,70$	$(33,70)^2 = 1135,35$
Kesembilan	264,36	$264,36 - 283,50 = -19,14$	$(-19,14)^2 = 366,53$
Kesepuluh	380,64	$380,64 - 283,50 = 97,13$	$(97,13)^2 = 9435,21$
Total	-	-	16692,16

Selanjutnya, hitung varians: $\sigma^2 = \frac{16692,16}{10-1} - \frac{16692,16}{9} = 1854,68 \text{ ms}^2$

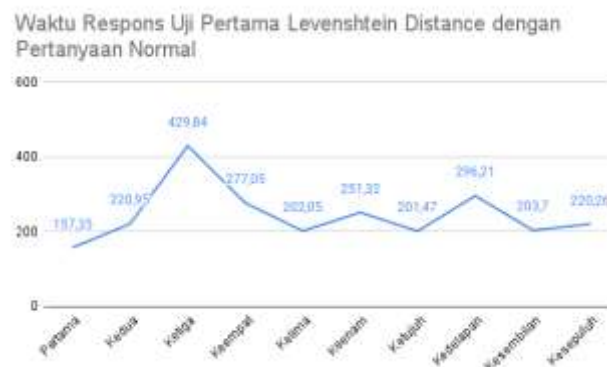
Lalu menghitung standar deviasi: $\sigma = \sqrt{\sigma^2} = \sqrt{1854,68} = 43,07 \text{ ms}$

Uji Pertama Levenshtein Distance dengan Pertanyaan Normal

Uji selanjutnya pada algoritma Levenshtein Distance pada pertanyaan normal:
 Input: “Bagaimana cara mendaftar menjadi mahasiswa?”



Gambar 10 Tangkapan Layar Kecepatan Menjawab Levenshtein Distance



Gambar 11 Waktu Menjawab Uji Keluaran Pertama Levenshtein Distance

waktu tercepat yang diperoleh adalah 157,35 ms, sedangkan waktu terlama adalah 429,84 ms, Selanjutnya hitung rata-ratanya:

$$\mu = \frac{157,35 + 220,95 + 429,84 + \dots + 220,26}{10} = 246,02 \text{ ms}$$

Tabel 2 Perhitungan Standar Deviasi Levenshtein Distance Pertama

Percobaan	Waktu Respons (ms)	$X_i - \mu$	$(X_i - \mu)^2$
Pertama	157,35	$157,35 - 246,02 = -88,67$	$(-88,67)^2 = 7862,37$
Kedua	220,95	$220,95 - 246,02 = -25,07$	$(-25,07)^2 = 628,50$
Ketiga	429,84	$429,84 - 246,02 = 183,82$	$(183,82)^2 = 33789,79$
Keempat	277,05	$277,05 - 246,02 = 31,03$	$(31,03)^2 = 962,86$
Kelima	202,05	$202,05 - 246,02 = -43,97$	$(-43,97)^2 = 1933,36$
Keenam	251,32	$251,32 - 246,02 = 5,30$	$(5,30)^2 = 28,09$
Ketujuh	201,47	$201,47 - 246,02 = -44,55$	$(-44,55)^2 = 1984,70$
Kedelapan	296,21	$296,21 - 246,02 = 50,19$	$(50,19)^2 = 2519,04$
Kesembilan	203,7	$203,70 - 246,02 = -42,32$	$(-42,32)^2 = 1790,98$
Kesepuluh	220,26	$220,26 - 246,02 = -25,76$	$(-25,76)^2 = 663,58$
Total	-	-	52163,28

Selanjutnya, hitung varians: $\sigma^2 = \frac{52163,28}{10-1} - \frac{52163,28}{9} = 5795,92 \text{ ms}^2$

Lalu menghitung standar deviasi: $\sigma = \sqrt{\sigma^2} = \sqrt{5795,92} = 76,13 \text{ ms}$

Hasil Uji Pertama pada Algoritma Ratcliff/Obershelp

Algoritma Ratcliff/Obershelp memiliki rata-rata waktu respons 283,50 ms dengan standar deviasi 40,37 ms, menunjukkan kestabilan yang lebih baik. Waktu eksekusi tercepatnya 234,35 ms dan terlama 380,64 ms.

Sebaliknya, algoritma Levenshtein Distance memiliki rata-rata waktu respons lebih cepat, yaitu 246,02 ms, tetapi dengan standar deviasi lebih tinggi (76,13 ms), menunjukkan fluktuasi yang lebih besar. Waktu tercepatnya 157,35 ms, sedangkan terlama 429,84 ms.

Kesimpulannya, Ratcliff/Obershelp lebih stabil dan konsisten, sedangkan Levenshtein Distance lebih cepat dalam beberapa kasus tetapi kurang stabil. Pemilihan algoritma tergantung pada prioritas antara kestabilan atau kecepatan.

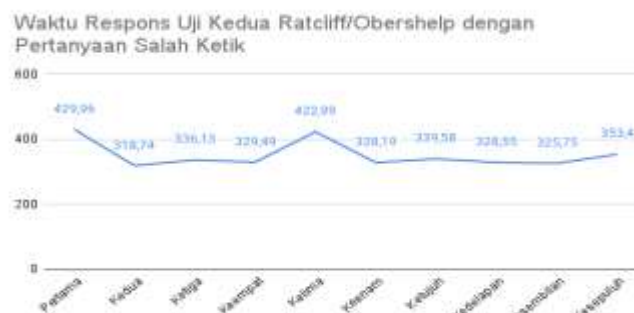
Uji Kedua Ratclif/Obershelp dengan Pertanyaan Salah Ketik

Uji Kedua Algoritma Ratcliff/Obershelp dengan pertanyaan salah ketik hasil rata-rata dan standar deviasi sebagai berikut:

Input: "Bsgimana cara mwndafrat mnjadu mhaiswa?"



Gambar 12 Tangkapan Layar Kecepatan Menjawab Ratcliff/Obershelp Kedua



Gambar 13 Waktu Menjawab Uji Kedua Ratcliff/Obershelp

waktu tercepat yang diperoleh adalah 318,74 ms, sedangkan waktu terlama adalah 429,96 ms, Selanjutnya hitung rata-ratanya:

$$\mu = \frac{429,96 + 318,74 + 336,13 + 329,49 + \dots + 353,40}{10} = 351,28 \text{ ms}$$

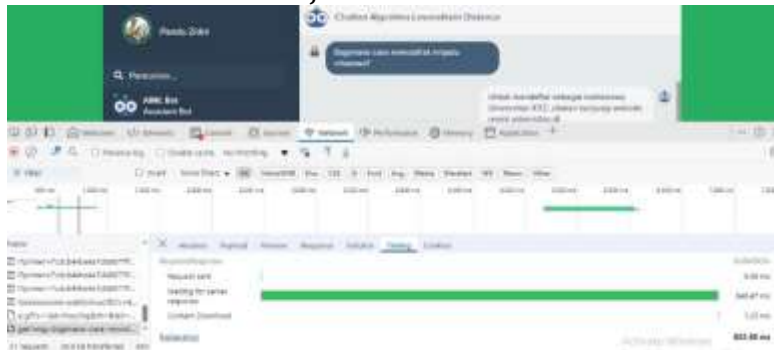
Tabel 3 Perhitungan Standar Deviasi Ratcliff/Obershelp Kedua

Percobaan	Waktu Respons	$X_i - \mu$	$(X_i - \mu)^2$
Pertama	429,96	$429,96 - 351,28 = 78,68$	$(78,68)^2 = 6190,86$
Kedua	318,74	$318,74 - 351,28 = -32,54$	$(-32,54)^2 = 1058,72$
Ketiga	336,13	$336,13 - 351,28 = -15,15$	$(-15,15)^2 = 229,46$
Keempat	329,49	$329,49 - 351,28 = -21,79$	$(-21,79)^2 = 474,72$
Kelima	422,99	$422,99 - 351,28 = 71,71$	$(71,71)^2 = 5142,61$
Keenam	328,19	$328,19 - 351,28 = -23,09$	$(-23,09)^2 = 533,06$
Ketujuh	339,58	$339,58 - 351,28 = -11,70$	$(-11,70)^2 = 136,84$
Kedelapan	328,55	$328,55 - 351,28 = -22,73$	$(-22,73)^2 = 516,56$
Kesembilan	325,75	$325,75 - 351,28 = -25,53$	$(-25,53)^2 = 651,68$
Kesepuluh	353,40	$353,40 - 351,28 = 2,12$	$(2,12)^2 = 4,50$
Total	-	-	14939,01

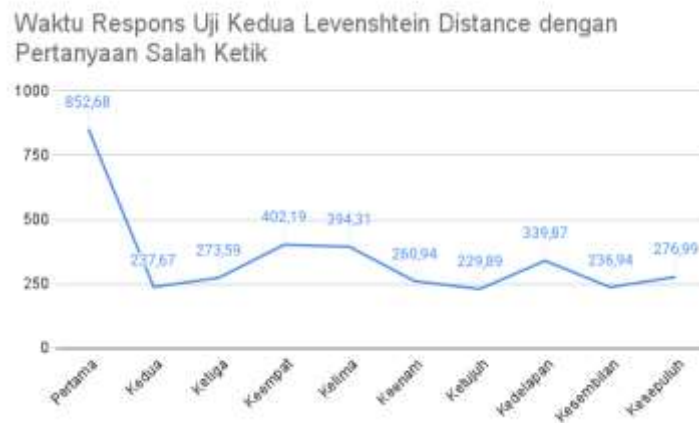
Selanjutnya, hitung varians: $\sigma^2 = \frac{14939.01}{10-1} - \frac{14939.01}{9} = 1659.89 \text{ ms}^2$
 Lalu menghitung standar deviasi: $\sigma = \sqrt{\sigma^2} = \sqrt{1659.89} = 40.74 \text{ ms}$

Uji Kedua Levenshtein Distance dengan Pertanyaan Salah Ketik

Sedangkan untuk Algoritma Levenshtein Distance dengan pertanyaan salah ketik hasil rata-rata dan standar deviasi sebagai berikut:
 Input: "Bsgimana cara mwndafrat mnjadu mhaiswa?"



Gambar 14 Tangkapan Layar Kecepatan Menjawab Levenshtein Distance Kedua



Gambar 15 Waktu Menjawab Uji Keluaran Kedua Levenshtein Distance

waktu tercepat yang diperoleh adalah 229,89 ms, sedangkan waktu terlama adalah 852,68 ms, Selanjutnya hitung rata-ratanya:

$$\mu = \frac{852.68 + 237.67 + 273.59 + \dots + 276.99}{10} = 350.51 \text{ ms}$$

Tabel 4 Perhitungan Standar Deviasi Levenshtein Distance Kedua

Percobaan	Waktu Respons (ms)	$X_i - \mu$	$(X_i - \mu)^2$
Pertama	852,68	852,68-350,51=502,17	$(502,17)^2=252177,72$
Kedua	237,67	237,67-350,51=-112,84	$(-112,84)^2=12732,19$
Ketiga	273,59	273,59-350,51=-76,92	$(-76,92)^2=5916,22$
Keempat	402,19	402,19-350,51=51,68	$(51,68)^2=2671,13$
Kelima	394,31	394,31-350,51=43,80	$(43,80)^2=1918,70$
Keenam	260,94	260,94-350,51=-89,57	$(-89,57)^2=8022,25$
Ketujuh	229,89	229,89-350,51=-120,62	$(-120,62)^2=14548,46$
Kedelapan	339,87	339,87-350,51=-10,64	$(-10,64)^2=113,15$
Kesembilan	236,94	236,94-350,51=-113,57	$(-113,57)^2=12897,46$
Kesepuluh	276,99	276,99-350,51=-73,52	$(-73,52)^2=5404,75$
Total	-	-	316402,04

Selanjutnya, hitung varians: $\sigma^2 = \frac{316402,04}{10-1} - \frac{316402,04}{9} = 35155.78 \text{ ms}^2$

Lalu menghitung standar deviasi: $\sigma = \sqrt{\sigma^2} = \sqrt{35155.78} = 187.50 \text{ ms}$

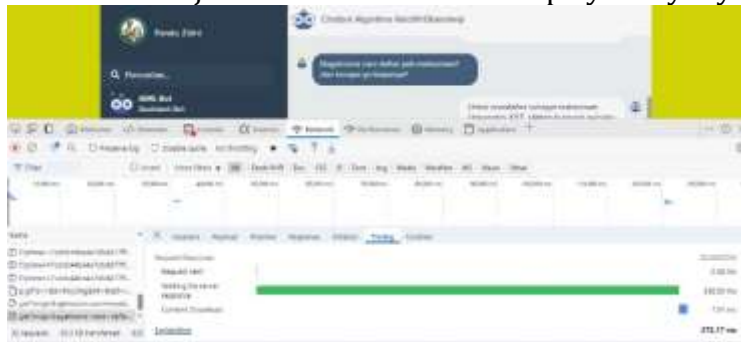
Hasil Uji Kedua pada Algoritma Ratcliff/Obershelp dan Levenshtein Distance

Chatbot dengan Ratcliff/Obershelp memiliki rata-rata respons 351,28 ms dengan standar deviasi kecil 40,74 ms, menunjukkan kestabilan dalam menangani kesalahan ketik. Levenshtein Distance lebih cepat dalam beberapa kondisi (rata-rata 350,51 ms, tetapi fluktuatif dengan standar deviasi tinggi 187,50 ms. Ratcliff/Obershelp lebih stabil, sementara Levenshtein Distance lebih cepat tetapi kurang konsisten, sehingga pemilihannya bergantung pada prioritas kestabilan atau kecepatan.

Uji Ketiga Ratclif/Obershelp dengan Pertanyaan Dalam Satu Masukkan

Selanjutnya untuk Algoritma Ratcliff/Obershelp dengan dua pertanyaan dalam satu input hasil rata-rata dan standar deviasi sebagai berikut:

Input: "Bagaimana cara daftar jadi mahasiswa? dan berapa ya biayanya?"



Gambar 16 Tangkapan Layar Kecepatan Menjawab Ratcliff/Obershelp Ketiga



Gambar 17 Waktu Menjawab Uji Ketiga Ratcliff/Obershelp

waktu tercepat yang diperoleh adalah 319,94 ms, sedangkan waktu terlama adalah 467,46 ms, Selanjutnya hitung rata-ratanya:

$$\mu = \frac{416.17 + 381.35 + 478.92 + \dots + 367.06}{10} = 397.12 \text{ ms}$$

Tabel 5 Perhitungan Standar Deviasi Ratcliff/Obershelp Ketiga

Percobaan	Waktu Respons	$\text{Xi}-\mu$	$(\text{Xi}-\mu)^2$
Pertama	327,17	$327.17-357.61=-30.44$	$(-30.44)^2=926.84$
Kedua	467,46	$467.46-357.61=109.85$	$(109.85)^2=12066.14$
Ketiga	362,42	$362.42-357.61=4.81$	$(4.81)^2=23.10$
Keempat	394,77	$394.77-357.61=37.16$	$(37.16)^2=1380.57$
Kelima	345,82	$345.82-357.61=-11.79$	$(-11.79)^2=139.10$
Keenam	368,51	$368.51-357.61=10.90$	$(10.90)^2=118.72$

Ketujuh	319,94	319.94-357.61=-37.67	$(-37.67)^2=1419.33$
Kedelapan	326,86	326.86-357.61=-30.75	$(-30.75)^2=945.81$
Kesembilan	323,23	323.23-357.61=-34.38	$(-34.38)^2=1182.26$
Kesepuluh	339,96	339.96-357.61=-17.65	$(-17.65)^2=311.66$
Total	-	-	18513.53

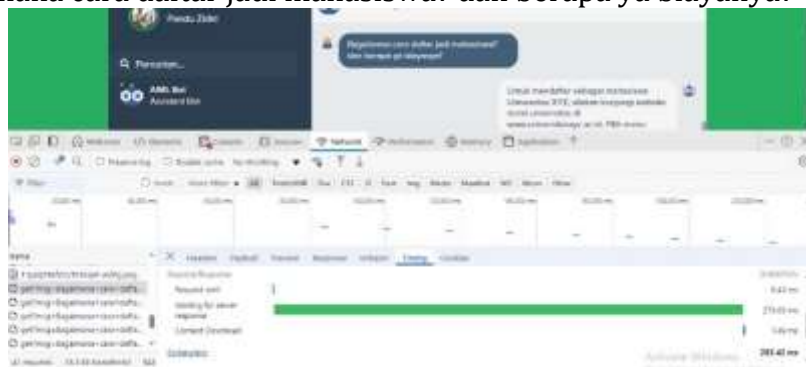
Selanjutnya, hitung varians: $\sigma^2 = \frac{18513.53}{10-1} - \frac{18513.53}{9} = 2057.06 \text{ ms}^2$

Lalu menghitung standar deviasi: $\sigma = \sqrt{\sigma^2} = \sqrt{2057.06} = 45.35 \text{ ms}$

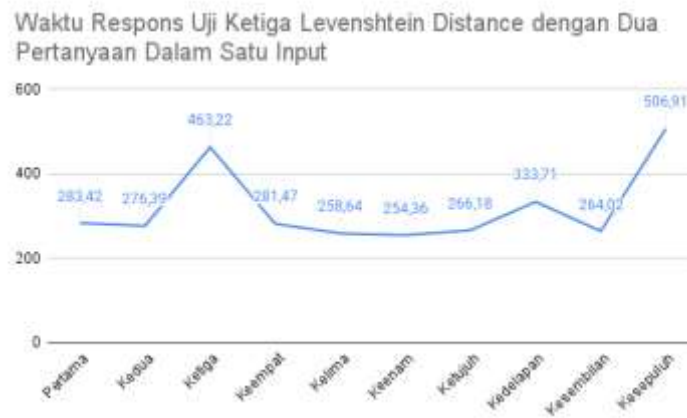
Uji Ketiga Levenshtein Distance dengan Pertanyaan Dalam Satu Masukkan

Dan yang terakhir yaitu pengujian algoritma Levenshtein Distance pada dua pertanyaan dalam satu input:

Input: "Bagaimana cara daftar jadi mahasiswa? dan berapa ya biayanya?"



Gambar 18 Tangkapan Layar Kecepatan Menjawab Levenshtein Distance Ketiga



Gambar 19 Waktu Menjawab Uji Keluaran Ketiga Levenshtein Distance

waktu tercepat yang diperoleh adalah 254,36 ms, sedangkan waktu terlama adalah 506,91 ms, Selanjutnya hitung rata-ratanya:

$$\mu = \frac{283.42 + 276.39 + 463.22 + \dots + 506.91}{10} = 318.83 \text{ ms}$$

Tabel 6 Perhitungan Standar Deviasi Levenshtein Distance Ketiga

Percobaan	Waktu Respons (ms)	$\text{Xi}-\mu$	$(\text{Xi}-\mu)^2$
Pertama	283,42	283,42-318,83=-35,41	$(-35,41)^2=1254,01$
Kedua	276,39	276,39-318,83=-42,44	$(-42,44)^2=1801,32$
Ketiga	463,22	463,22-318,83=144,39	$(144,39)^2=20847,89$
Keempat	281,47	281,47-318,83=-37,36	$(-37,36)^2=1395,92$
Kelima	258,64	258,64-318,83=-60,19	$(-60,19)^2=3623,08$
Keenam	254,36	254,36-318,83=-64,47	$(-64,47)^2=4156,64$
Ketujuh	266,18	266,18-318,83=-52,65	$(-52,65)^2=2772,23$

Kedelapan	333,71	333,71-318,83=14,88	(14,88) ² =221,35
Kesembilan	264,02	264,02-318,83=-54,81	(-54,81) ² =3004,36
Kesepuluh	506,91	506,91-318,83=188,08	(188,08) ² =35373,33
Total	-	-	74450.14

Selanjutnya, hitung varians: $\sigma^2 = \frac{74450.14}{10-1} - \frac{74450.14}{9} = 8272.24 \text{ ms}^2$

Lalu menghitung standar deviasi: $\sigma = \sqrt{\sigma^2} = \sqrt{8272.24} = 90.95 \text{ ms}$

Hasil Uji Ketiga pada Algoritma Ratcliff/Obershelp

Ratcliff/Obershelp lebih stabil 357,61 ms, deviasi 45,35 ms dalam menangani dua pertanyaan dalam satu input, sementara Levenshtein Distance lebih cepat 318,83 ms tetapi kurang konsisten deviasi 90,95 ms.

Kesimpulannya, Ratcliff/Obershelp lebih stabil untuk menangani dua pertanyaan dalam satu input, sedangkan Levenshtein Distance lebih cepat tetapi kurang konsisten.

Rangkuman Hasil Analisa Algoritma Ratcliff/Obershelp dengan Levenshtein Distance

Secara keseluruhan, Ratcliff/Obershelp lebih stabil dan konsisten dalam menangani kesalahan ketik serta input dengan banyak pertanyaan, sementara Levenshtein Distance lebih cepat tetapi kurang stabil. Jika kestabilan diutamakan, Ratcliff/Obershelp lebih unggul, sedangkan Levenshtein Distance cocok untuk kecepatan dengan risiko fluktuasi.

KESIMPULAN

Penelitian ini berhasil mengembangkan chatbot AIML untuk layanan informasi akademik di Universitas XYZ. Dengan algoritma Ratcliff/Obershelp dan Levenshtein Distance, chatbot dapat mencocokkan pertanyaan mahasiswa secara efektif. Ratcliff/Obershelp lebih toleran terhadap kesalahan ketik, sementara Levenshtein Distance lebih cepat dalam mencocokkan pola terstruktur. Chatbot ini meningkatkan efisiensi pencarian informasi dan mendukung digitalisasi kampus, meskipun masih menghadapi tantangan dalam variasi waktu eksekusi dan keterbatasan menangani pertanyaan di luar database AIML. Pengembangan lebih lanjut diperlukan untuk mendukung pembelajaran otomatis.

DAFTAR PUSTAKA

- [1] G. Guntoro, Loneli Costaner, and L. Lisnawita, "Aplikasi Chatbot untuk Layanan Informasi dan Akademik Kampus Berbasis Artificial Intelligence Markup Language (AIML)," *Digit. Zo. J. Teknol. Inf. dan Komun.*, vol. 11, no. 2, pp. 291-300, 2020, doi: 10.31849/digitalzone.v11i2.5049.
- [2] A. Z. Amrullah, A. S. Anas, and G. Primajati, "Implementasi Chatbot sebagai Virtual Assistant Penerimaan Mahasiswa Baru pada Universitas Bumigora," *J. Bumigora Inf. Technol.*, vol. 4, no. 1, pp. 17-26, 2022, doi: 10.30812/bite.v4i1.1664.
- [3] A. N. Aqil, B. Dirgantara, Istikmal, U. A. Ahmad, and R. R. Septiawan, "Robot Chat System (Chatbot) To Help Users 'Homelab' Based In Deep Learning," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 8, pp. 599-604, 2021, doi: 10.14569/IJACSA.2021.0120870.
- [4] T.-J. Goh, L.-Y. Chong, S.-C. Chong, and P.-Y. Goh, "A Campus-based Chatbot

- System using Natural Language Processing and Neural Network,” *J. Informatics Web Eng.*, vol. 3, no. 1, pp. 96–116, 2024, doi: 10.33093/jiwe.2024.3.1.7.
- [5] D. M. Park, S. S. Jeong, and Y. S. Seo, “Systematic Review on Chatbot Techniques and Applications,” *J. Inf. Process. Syst.*, vol. 18, no. 1, pp. 26–47, 2022, doi: 10.3745/JIPS.04.0232.
- [6] 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 2017*, 2017. doi: 10.1109/ICACCI.2017.8126057.
- [7] O. A. S. Ibrahim, B. A. Hamed, and T. A. El-Hafeez, “A new fast technique for pattern matching in biological sequences,” *J. Supercomput.*, vol. 79, no. 1, pp. 367–388, 2023, doi: 10.1007/s11227-022-04673-3.
- [8] V. Christanti, J. Jesslyn, and F. Orlando, “Implementasi Chatbot Pelajaran Sekolah Dasar Dengan Pandorabots,” *Cices*, vol. 9, no. 2, pp. 203–213, 2023, doi: 10.33050/cices.v9i2.2703.
- [9] T. L. M. Suryanto, A. P. Wibawa, Hariyono, and A. Nafalski, “Evolving Conversations: A Review of Chatbots and Implications in Natural Language Processing for Cultural Heritage Ecosystems,” *Int. J. Robot. Control Syst.*, vol. 3, no. 4, pp. 955–1006, 2023, doi: 10.31763/ijrcs.v3i4.1195.
- [10] T. Szép, S. van Cranenburgh, and C. Chorus, “Moral rhetoric in discrete choice models: a Natural Language Processing approach,” *Qual. Quant.*, vol. 58, no. 1, pp. 179–206, 2024, doi: 10.1007/s11135-023-01625-8.
- [11] M. Affandes and P. Pizaini, “Academic Information Service Chatbot Using HMM and AIML,” *J. CoreIT J. Has. Penelit. Ilmu Komput. dan Teknol. Inf.*, vol. 8, no. 2, p. 79, 2022, doi: 10.24014/coreit.v8i2.19638.
- [12] J. J. Bird, A. Ekárt, and D. R. Faria, “Chatbot Interaction with Artificial Intelligence: human data augmentation with T5 and language transformer ensemble for text classification,” *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 4, pp. 3129–3144, 2023, doi: 10.1007/s12652-021-03439-8.
- [13] M. Muzakkir, R. Juliana, and M. Murhaban, “Pelatihan Penulisan Feature Jurnalistik Bagi Calon Insan Pers pada Harian Serambi Indonesia,” *J. Pengabd. Masy. Darma Bakti Teuku Umar*, vol. 3, no. 2, p. 166, 2022, doi: 10.35308/baktiku.v3i2.4362.
- [14] M. A. Rosid, A. S. Fitriani, I. R. I. Astutik, N. I. Mulloh, and H. A. Gozali, “Improving Text Preprocessing for Student Complaint Document Classification Using Sastrawi,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 874, no. 1, 2020, doi: 10.1088/1757-899X/874/1/012017.
- [15] N. Izzah, N. Yusliani, and D. Roodiah, “Sistem Deteksi Kemiripan Teks Pada Berita Berbahasa Indonesia Menggunakan algoritma Ratcliff/Obershelp,” *J. Linguist. Komputasional*, vol. 5, no. 1, p. 1, 2022, doi: 10.26418/jlk.v5i1.65.
- [16] T. N. Son, V. T. Nhan, L. Minh, L. M. Thanh, N. Van Phuc, and D. P. H. Trang, “A chatbot using Levenshtein distance algorithm for Raspberry board,” *J. Tech. Educ. Sci.*, vol. 15, no. 6, pp. 55–61, 2020, [Online]. Available: <https://jte.edu.vn/index.php/jte/article/view/27>
- [17] S. Arsovski, M. I. Oladele, A. D. Cheok, V. Premcevski, and B. Markoski, “An Approach to Email Categorization and Response Generation,” *Comput. Sci. Inf. Syst.*, vol. 19, no. 2, pp. 913–934, 2022, doi: 10.2298/CSIS211101009A.

- [18] Maukar, E. Sutanty, R. Arianty, and E. Setyaningsih, “Implementasi Algoritma Similaritas Dokumen Ratcliff / Obershelp pada Pendeteksian Similaritas Dokumen,” vol. 4, no. 3, pp. 844–853, 2024.
- [19] J. Kawijaya, “Penggunaan Artificial Intelligence Markup Language (AIML) Untuk Menganalisa Kesalahan Menulis Bahasa Arab di MTs Bilingual Batu,” *J. Jendela Pendidik*, vol. 3, no. 03, pp. 352–362, 2023, doi: 10.57008/jjp.v3i03.569.
- [20] S. Debora Ina, S. D.I. Mau, and M. Wilda Malo, “Penerapan Aplikasi Pengelolaan Data Siswa Smp Tutim Delo Dengan Menggunakan Metode System Development Life Cycle (Sdlc),” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 5, pp. 3806–3810, 2024, doi: 10.36040/jati.v7i5.7948.
- [21] P. A. Nugroho and D. Hernandi, “Perancangan Sistem Informasi Untuk Penyewaan Jasa Fotografi Berbasis Web Pada Appa Project,” *Jris J. Rekayasa Inf. Swadharma*, vol. 4, no. 1, pp. 10–17, 2024, doi: 10.56486/jris.vol4no1.399.
- [22] M. A. Rasubala, M. S. C. Kaunang, and S. Sunarmi, “Penerapan Model Problem Based Learning (Pbl) Pada Pembelajaran Seni Musik Vokal Di Kelas Viii Smp Negeri 4 Tondano,” *Kompetensi*, vol. 3, no. 03, pp. 2130–2142, 2023, doi: 10.53682/kompetensi.v3i03.6026.