

PENGEMBANGAN CHATBOT TANYA JAWAB YANG INDEPENDEN UNTUK DISCORD MENGGUNAKAN *NATURAL LANGUAGE PROCESSING* DAN *MACHINE LEARNING*

Ferrary Wijaya¹, Nanda Goaw Putra², Michael³

Program Studi Teknik Informatika

Universitas Tarumanagara

Jakarta, Indonesia

e-mail: ¹ferrary.535220254@stu.untar.ac.id, ²nanda.535220069@stu.untar.ac.id,

³michael.535220261@stu.untar.ac.id

Correspondence : e-mail: vinym@fti.untar.ac.id

Diajukan:; Direvisi:; Diterima:

Abstrak

Chatbot telah menjadi alat penting dalam interaksi manusia-komputer, terutama di platform seperti Discord. Pengembangan chatbot yang mampu memberikan jawaban informatif secara cerdas semakin krusial untuk meningkatkan pengalaman pengguna dan memberikan nilai tambah bagi komunitas online. Penelitian ini bertujuan untuk mengembangkan chatbot tanya jawab untuk Discord menggunakan teknologi NLP (*Natural Language Processing*) dan ML (*Machine Learning*). Chatbot ini dirancang untuk memahami pertanyaan dalam bahasa alami, memprosesnya dengan teknik seperti tokenisasi, stemming, lemmatization, dan part-of-speech tagging, serta menghasilkan jawaban yang relevan menggunakan model ML berbasis klasifikasi dan regresi. Metode penelitian melibatkan kombinasi NLP dan ML dengan melatih chatbot menggunakan data pertanyaan dan jawaban terkait Discord, serta memanfaatkan basis pengetahuan yang diperbarui secara berkala. Hasil penelitian menunjukkan bahwa chatbot yang dikembangkan dapat dengan akurat memahami dan merespons pertanyaan pengguna. Fungsinya mencakup memberikan informasi tentang Discord, menangani masalah teknis, dan menyediakan dukungan komunitas. Pengembangan ini membuka peluang untuk meningkatkan interaksi dan kolaborasi dalam komunitas Discord, memberikan layanan yang lebih personal dan efektif kepada pengguna, serta berpotensi meningkatkan engagement dan retensi pengguna.

Kata kunci: Discord Chatbot, ML, NLP, Tanya jawab.

Abstract

Chatbots have become essential tools in human-computer interaction, particularly on platforms like Discord. Developing a chatbot capable of providing intelligent and informative answers is increasingly crucial for enhancing user experience and adding value to online communities. This research aims to build a question-answering chatbot for Discord using NLP (*Natural Language Processing*) and ML (*Machine Learning*) technologies. The chatbot is designed to understand natural language questions, process them using techniques such as tokenization, stemming, lemmatization, and part-of-speech tagging, and generate relevant answers using ML models based on classification and regression. The research methodology combines NLP and ML by training the chatbot on a dataset of Discord-related questions and answers, supported by regularly updated knowledge bases. The results show that the developed question-answering chatbot accurately understands and responds to user inquiries. Its functions include providing information about Discord, addressing technical issues, and offering community support. This development opens opportunities to enhance interaction and collaboration within Discord communities, providing more personalized and effective services to users, and potentially boosting user engagement and retention.

Keywords: Discord Chatbot, Machine Learning, Natural Language Processing, Question and answer.

1. Pendahuluan

Chatbot adalah sebuah program komputer yang dirancang untuk mensimulasikan sebuah percakapan atau komunikasi yang interaktif kepada pengguna (manusia) melalui bentuk teks, suara, dan atau visual. Percakapan yang terjadi antara komputer dengan manusia merupakan bentuk respon dari program yang telah dideklarasikan pada database program pada komputer. Kemampuan komputer dalam menyimpan banyaknya data tanpa melupakan satu pun informasi yang disimpannya digabungkan dengan kepraktisan bertanya pada sumber informasi langsung dibandingkan dengan mencari informasi sendiri serta kemampuan learning yang dimilikinya menyebabkan *chatbot* adalah customer service yang handal.

Respon yang dihasilkan merupakan hasil pemindaian kata kunci pada inputan pengguna dan menghasilkan respon balasan yang dianggap paling cocok, atau pola kata-kata yang dianggap paling mendekati dan pada umumnya menggunakan pendekatan *Natural Language Processing (NLP)*. *NLP* merupakan salah satu tujuan jangka panjang dari *Artificial Intelligence* (kecerdasan buatan) yaitu pembuatan program yang memiliki kemampuan untuk memahami bahasa manusia. Pada prinsipnya bahasa alami adalah suatu bentuk representasi dari suatu pesan yang ingin dikomunikasikan antar manusia. Bentuk utama representasinya adalah berupa suara/ucapan, tetapi sering pula dinyatakan dalam bentuk tulisan.

NLP pada aplikasi berkaitan dengan bagaimana komputer dapat digunakan untuk memahami dan memanipulasi teks bahasa alami (*Natural Language*) untuk mendapatkan informasi tertentu. Dengan perantara bahasa alami (*Natural Language*) inilah, manusia dapat berinteraksi dengan komputer. *NLP* digunakan dalam pemrosesan dokumen karena user menentukan relevansi dari dokumen dengan membaca dan menganalisisnya. Jika sistem dapat melakukan analisis dokumen secara otomatis, maka proses pencarian dokumen yang relevan akan lebih mudah.

Salah satu tempat dimana *chatbot* banyak bermunculan berada pada salah satu platform terkemuka yaitu Discord. Discord telah menjadi platform komunikasi dan komunitas online yang berkembang pesat, terutama sejak tahun 2020 atau awal pandemi *COVID-19* yang meningkat sebanyak 194 juta pengguna hingga tahun 2024. Platform ini juga menawarkan berbagai fitur seperti pesan teks, suara, video, memungkinkan pengguna untuk saling terhubung, berinteraksi, dan berkolaborasi dengan mudah.

Pertumbuhan Discord yang pesat telah meningkatkan kebutuhan akan solusi komunikasi yang lebih efektif dan efisien. *Chatbot* tanya jawab menawarkan solusi yang menjanjikan untuk mengatasi beberapa tantangan yang dihadapi pengguna Discord. Selain dari *chatbot* tanya-jawab Discord juga memiliki bot-bot lainnya seperti bot trading harga barang (*growstocks*), bot music (*rhythm*, *fredbot*, *luna*) dan lainnya. Harapan kami dengan adanya penelitian pengembangan ini dapat menjadi paduan atau *guide* untuk pembaca dan juga penulis demi mengembangkan sebuah *chatbot* yang dapat menjawab berbagai pertanyaan pengguna secara komprehensif dan informatif, tanpa bergantung pada intervensi manusia.

2. Metode Penelitian

Metode penilitan yang akan kami gunakan meliputi beberapa level dari NLP itu sendiri yaitu metode tokenisasi, *stemming*, *lemmatization*, *part-of-speech-tagging* dan menggunakan metode dasar yang di ambil dari *ML* yaitu klasifikasi dan regresi. Dimana metode dasar NLP itu sendiri bertujuan agar bot mengerti segala bentuk perkataan yang dilontarkan oleh pengguna dan metode dasar *ML* yang memungkinkan komputer untuk belajar dari data tanpa deprogram secara eksplisit:

1. Tokenisasi

Metode awal yang akan digunakan yaitu tokenisasi dimana teks atau string input akan diproses dan dipotong berdasarkan kata-kata yang menyusunnya. Tokenisasi menghasilkan seluruh kata-kata yang telah dipisahkan dalam suatu kalimat. Pembersihan Data (*Data Cleaning*).

2. *Stemming and Lemmatization*

Stemming dan *lemmatization* salah satu metode dari NLP itu sendiri yang akan kita gunakan dimana *stemming* berfungsi untuk mengurangi kata-kata ke bentuk dasarnya dengan menghilangkan akhiran dan awalan, dan *lemmatization* berperan sebagai *autocorrect* dimana fungsi ini akan mengurangi kata-kata ke bentuk kamus yang tepat dengan mempertimbangkan konteksnya [4].

3. *Part Of Speech Tagging*

Setelah *string input* dari *user* sudah melewati 3 tahap di atas, di tahap ini *string input* tersebut akan di indentifikasi jenis katanya apakah kata tersebut masuk ke golongan (kata benda, kata kerja, dll) dalam sebuah kalimat.

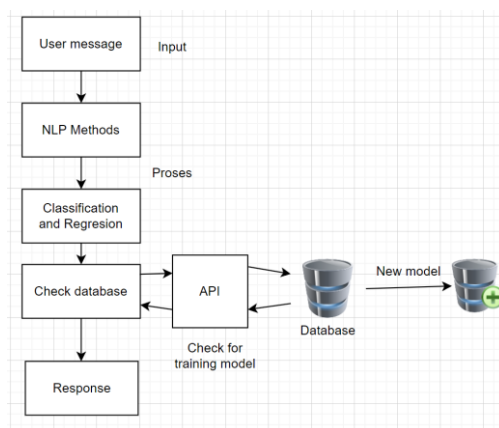
4. Klasifikasi

Setelah kata-kata tersebut di proses melalui 3 metode NLP di atas, metode selanjutnya adalah klasifikasi yang merupakan metode pembelajaran mesin yang diawasi dimana model mencoba memprediksi label yang benar dari data masukan tertentu. Misalnya, suatu program dapat memprediksi apakah email merupakan email spam atau bukan melalui data pelatihan yang telah diberikan.

5. Regresi

Metode regresi itu sendiri memiliki beberapa tipe yaitu (regresi logistik, regresi linear, regresi polinomial, regresi lidge dan lasso, dan regresi *random forest*) merupakan teknik dimana program dilatih untuk memprediksi nilai dari sebuah variable berdasarkan *inputan* yang diberikan, Jadi setelah *string input* diproses dan diklasifikasikan melalui metode sebelumnya, *string input* tersebut akan dicocokkan dengan data yang ada sehingga sistem dapat memberikan jawaban atas pertanyaan pengguna.

1.1 Alur Kerja Chatbot



Gambar 1. Alur kerja *chatbot*.

Gambar 1 menjelaskan secara kasar alur kerja *chatbot* yang akan dirancang dimana pertama bot akan menerima *user message/ user input* kemudian teks tersebut akan melewati 2 proses terlebih dahulu yaitu proses NLP dan ML, dimana kalimat *input* akan diklasifikasikan kemudian kalimat tersebut akan dicocokkan dengan database melalui API, dan jika kalimat atau pertanyaan yang diberikan oleh *user* ada di *database training* maka *chatbot* akan langsung memberikan response berupa rangkain kata-kata, namun apabila pertanyaan yang diberikan *user* belum ada masuk ke dalam *database* maka pertanyaan tersebut akan dimasukkan kedalam *database* pertanyaan yang khusus untuk menyimpan list pertanyaan yang belum ada di *data training* dan kemudian memberikan response kembali kepada *user*.

3. Hasil dan Pembahasan

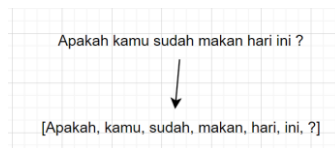
Penelitian ini bertujuan untuk mengembangkan *chatbot* tanya jawab untuk Discord menggunakan NLP dan ML. *Chatbot* ini diharapkan dapat menjawab berbagai pertanyaan penggunaan secara komprehensif dan informatif, tanpa bergantung pada intervensi manusia[1].

Teknik NLP yang digunakan dalam penelitian ini meliputi tokenisasi, stemming, lemmatization, dan part-of-speech tagging. Teknik-teknik ini membantu *chatbot* untuk memahami struktur dan makna kalimat pengguna. Model ML yang digunakan dalam penelitian ini adalah model klasifikasi dan regresi. Model klasifikasi digunakan untuk mengkategorikan pertanyaan pengguna berdasarkan maksudnya, sedangkan model regresi digunakan untuk memprediksi jawaban atas pertanyaan pengguna berdasarkan data pelatihan.

3.1. Tahapan Processing

3.1.1 Tokenisasi

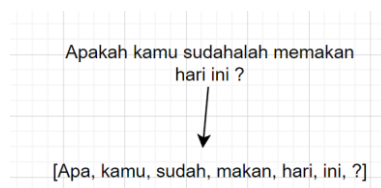
Tokenisasi dilakukan untuk mengurai teks menjadi kata-kata individual, yang kemudian dapat diproses lebih lanjut untuk memahami makna dari pertanyaan yang diberikan pengguna [3].



Gambar 2. Tokenisasi

3.1.2 Stemming

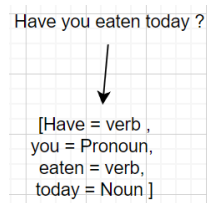
Stemming dilakukan untuk mengurangi variasi kata yang memiliki akar yang sama sehingga kata-kata tersebut dapat dianggap sebagai bentuk yang sama, meskipun ada variasi ejaan atau penggunaan [4].



Gambar 3. *Stemming*

3.1.3 Part-Of-Speech Tagging (POS TAGGING)

Part-of-speech tagging (POS Tagging) adalah proses pemberian label pada setiap kata dalam sebuah kalimat berdasarkan kelas kata, seperti kata benda, kata kerja, kata sifat, kata keterangan, dan lain-lain. Label ini membantu computer untuk memahami struktur dan makna kalimat [9].



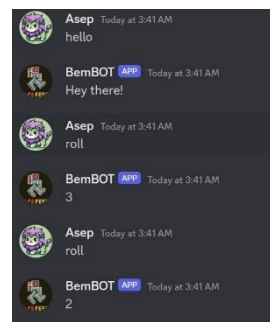
Gambar 4. *Pos Tagging*

3.1.4 Klasifikasi

Klasifikasi adalah proses mengelompokkan data ke dalam kategori atau kelas yang telah ditentukan sebelumnya. Dalam konteks machine learning, klasifikasi merupakan tugas yang diajarkan kepada model machine learning untuk memprediksi kelas mana yang paling sesuai untuk sebuah data baru [1].

```

1  import random
2
3  def get_response ( message: str) -> str :
4      p_message = message.lower()
5
6      if p_message == 'hello' :
7          return 'Hey there!'
8
9      if message == 'roll' :
10         return str(random.randint(1,6))
11
12     if p_message == 'help':
13         return 'Modify'
14
15     return 'I didnt understand what you wrote '
  
```



Gambar 5. Data Train dan Hasil Klasifikasi

Data training yang ditampilkan pada Gambar 5 dimasukkan secara manual (belum otomatis) disini kami masih belum mengimplementasikan metode kalsifikasi secara otomatis dimana bot langsung membaca *data set* yang diberikan, kemudian di gambar 6 merupakan contoh langsung dari implementasi klasifikasi, Dimana “Asep” berkata ‘hello’ kemudian BemBOT membaca *data* tersebut dan mencari apakah kata-kata tersebut ada didalam sistem, dan jika ada maka BemBOT akan memberikan balasan sesuai dengan sistem yaitu “Hey there!” [1]-[3], [5]-[10].

3.2. Pembahasan

Penelitian ini menunjukkan bahwa pengembangan *chatbot* tanya jawab untuk Discord menggunakan metode dasar NLP dan ML merupakan sebuah penelitian yang menjanjikan. Dan pengembangan *chatbot* ini memiliki beberapa manfaat bagi pengguna juga bagi pengembang aplikasi itu sendiri.

Chatbot Discord ini dilatih dengan 5 pertanyaan umum yang tersaji pada Tabel 1, untuk meningkatkan kemampuannya dalam memberikan respons yang relevan kepada pengguna. Meskipun demikian, *chatbot* ini masih menghadapi tantangan dalam menangani pertanyaan baru yang tidak termasuk dalam dataset yang telah dipelajarinya. Kelemahan utamanya adalah dalam hal keterbatasan untuk mengenali dan merespons pertanyaan yang tidak sesuai dengan pola yang telah dikenalnya.

Chatbot memiliki persyaratan ketat terhadap format pengetikan. Hal ini berarti pengguna harus memastikan bahwa pertanyaan atau pernyataan mereka diketik dengan benar, termasuk dalam hal penggunaan huruf besar-kecil dan struktur kalimat yang sesuai. Ketidaksesuaian dalam format penulisan dapat mengakibatkan *chatbot* gagal memahami pertanyaan atau memberikan jawaban yang tidak relevan. Tantangan lainnya adalah ketiadaan fitur *auto-correction* yang dapat mengoreksi kesalahan penulisan pengguna secara otomatis. Pengguna harus memastikan bahwa setiap kata dan kalimat yang mereka masukkan sesuai dengan aturan yang telah diajarkan kepada *chatbot* agar dapat berfungsi dengan optimal. Meskipun penelitian ini memiliki beberapa keterbatasan, penelitian ini menunjukkan bahwa *chatbot* yang dikembangkan dalam penelitian ini dapat menjadi alat yang berharga untuk meningkatkan pengalaman pengguna dan memperkuat komunitas Discord.

Tabel 1. Data pertanyaan umum *Chatbot* Discord

No	Pertanyaan	Jawaban	Keterangan
1	Hello, hi, how are you	Hey There!, Hello, Good to see you again	Terjawab
2	name, your name, what are you called	I'm Mind Reader, I am a Chatbot, I am your helper	Terjawab
3	cya, see you later, goodbye	Sad to see you go :(, Talk to you later, Goodbye	Terjawab
4	who created you, developers, creators	College students	Terjawab
5	What is progamming, What is coding, Tell me about programming	Programming, coding or software development, means writing computer code to automate tasks	Terjawab

Berdasarkan pertanyaan yang dimasukkan kedalam *chatbot*, seluruh pertanyaan berhasil terjawab sesuai dengan *command* yang diatur oleh penulis.

3.3. Harapan

Harapan dari penilitan ini, penulis sekaligus pengembang *chatbot* Discord juga memiliki harapan untuk peneliti berikutnya, yakni:

1. Peningkatan jumlah dan keragaman data pelatihan.
2. Mengembangkan model ML yang lebih canggih.
3. Mengebangkan teknik NLP yang lebih canggih untuk memahami konteks percakapan.
4. Peningkatan akurasi dan keefektifak *chatbot*
5. Perluasan fungsi dan metode untuk *chatbot*

4. Kesimpulan

Pengembangan *chatbot* tanya jawab di Discord menggunakan NLP dan ML didasarkan pada teori-teori yang kuat dari berbagai bidang komputer. Teori-teori ini membantu memastikan bahwa *chatbot* dapat memahami, memproses dan menjawab pertanyaan pengguna secara efisien. Namun dengan dasar teori yang telah kami coan implementasikan menyatakan bahwa masih jauh dari kata sempurna bila ingin membuat *chatbot* yang bisa bergerak secara indepen untuk menanggapi berbagai pertanyaan yang diluar dari *dataset* yang telah diberikan. Dengan adanya makalah ini kami harap dapat membantu pembaca maupun pembuat agar dapat meniliti dan mengembangkan lebih lanjut tentang *chatbot* yang akan dibuat di masa depan nanti.

Daftar Pustaka

- [1] D. Ramu, R. Jain, and A. Jain, "Generation Z's Ability to Discriminate Between AI-generated and Human-Authored Text on Discord," 2023, [Online]. Available: <http://arxiv.org/abs/2401.04120>
- [2] H. Singh and A. Agnihotri, "Discord Bot," 2022, [Online]. Available: <http://www.ir.juit.ac.in:8080/jspui/handle/123456789/3628>
- [3] D. Kzyrov and S. Sapakova, "Chatbot Assistants: Implementation and Analysis of the Efficiency," *J. Probl. Comput. Sci. Inf. Technol.*, vol. 1, no. 2, pp. 25–41, 2023, doi: 10.26577/jpcsit.2023.v1.i2.03.
- [4] M. Aleedy, H. Shaiba, and M. Bezbradica, "Generating and Analyzing Chatbot Responses using Natural Language Processing," *IJACSA Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 9, 2019, [Online]. Available: www.ijacsa.thesai.org
- [5] P. Anki, A. Bustamam, H. S. Al-Ash, and D. Sarwinda, "Intelligent Chatbot Adapted from Question and Answer System Using RNN-LSTM Model," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Mar. 2021. doi: 10.1088/1742-6596/1844/1/012001.
- [6] M. Nuruzzaman and O. K. Hussain, "Identifying facts for chatbot's question answering via sequence labelling using recurrent neural networks," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, May 2019. doi: 10.1145/3321408.3322626.
- [7] R. Kumar and M. M. Ali, "A Review on Chatbot Design and Implementation Techniques," *Int. Res. J. Eng. Technol.*, 2020, [Online]. Available: www.irjet.net
- [8] J. Edu, C. Mulligan, F. Pierazzi, J. Polakis, G. Suarez-Tangil, and J. Such, "Exploring the Security and Privacy Risks of Chatbots in Messaging Services," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference, IMC*, Association for Computing Machinery, Oct. 2022, pp. 581–588. doi: 10.1145/3517745.3561433.
- [9] A. Nugroho, D. Pramono Adi, and A. Bimo Gumelar, "Chatbot Untuk Customer Service Berbasis Teks dan Suara pada Sistem Manajemen Pemesanan (OMS) Menggunakan Platform Android," *REPOSITOR*, vol. 2, no. 6, pp. 683–690, 2020.
- [10] N. A. M. Herwanza, N. S. Harahap, F. Yanto, and F. Insani, "Penerapan Langchain Retriever dengan Model Chat Openai dalam Pengembangan Sistem Chatbot Hadis Berbasis Telegram," *JTIM J. Teknol. Inf. dan Multimed.*, vol. 6, no. 1, pp. 70–83, May 2024, doi: 10.35746/jtim.v6i1.514.