

## APLIKASI CHATBOT BERBASIS TEKS MENGGUNAKAN ALGORITMA NAIVE BAYES CLASSIFIER FAQ GRABADS

Rena Cahya Utama<sup>1</sup>, Fauziah<sup>2</sup>, Ratih Titi Komalasari<sup>3</sup>

Program Studi Teknik Informatika, Fakultas Teknologi Komunikasi dan Informatika, Universitas Nasional

rnchyhtm@gmail.com<sup>1</sup>, fauziah@civitas.unas.ac.id<sup>2</sup>, ratih.titi@civitas.unas.ac.id<sup>3</sup>

*Submitted June 14, 2021; Revised June 14, 2021; Accepted August 1, 2021*

### Abstrak

Hampir seluruh aspek dalam hidup manusia sudah tersentuh oleh perkembangan teknologi komunikasi dan informasi. Apalagi di masa pandemi yang terjadi saat ini dimana interaksi tatap muka sangatlah terbatas sehingga terkadang informasi tidak tersampaikan secara jelas dan tepat. Misalnya dalam hal pekerjaan ketika ada pegawai baru yang kesulitan ketika ingin mendapatkan informasi atau pun sekedar bertanya kepada teman kerja karna tidak bisa menyampaikan informasi dengan jelas. Minimnya akses yang dimiliki dan sumber informasi yang berbeda beda untuk mendapatkan informasi tertentu membuat para pegawai baru kesulitan dalam bekerja padahal pertanyaan yang ditanyakan adalah pertanyaan yang sama dengan yang ditanyakan oleh pegawai satu dan lainnya. Chatbot adalah hasil inovasi teknologi yang bertujuan untuk mempermudah proses pertukaran informasi dengan menggunakan program interaktif. Dengan aplikasi Chatbot berbasis teks ini dapat meminimalisir waktu yang diperlukan untuk menjawab pertanyaan karena sudah terotomatisasi. Algoritma Naive Bayes Classifier digunakan dalam pembuatan Chatbot dalam penelitian ini dengan sebelumnya dibuat kumpulan pertanyaan yang sering timbul (FAQ) berisi 10 pertanyaan beserta jawabannya yang akan dijadikan sebagai dataset. Dengan menggunakan split ratio sebesar 0,8 dan total 60 pertanyaan maka dihasilkan nilai akurasi sebesar 93,33 % dan nilai kesalahan sebesar 6,66 %.

**Kata Kunci :** Algoritma, aplikasi, chatbot, faq, naive bayes classifier

### Abstract

*Almost all aspects of human life has been touched by the development of information and communication technology. Especially in the current pandemic where face-to-face interactions are very limited, so sometimes information is not conveyed clearly and precisely. For example, in the case of work when there are new employees who have difficulty getting information or just asking colleagues because they cannot convey information clearly. The lack of access and the amount of access needed to obtain certain information makes it difficult for new employees to work even though the questions asked are the same questions asked by one employee or another. Chatbot is the result of technological innovation that aims to facilitate the process of exchanging information by using interactive programs. With this text-based Chatbot application, you can minimize the time needed to answer questions because it is automated. The Naive Bayes Classifier algorithm was used in making Chatbot in this study by previously creating a collection of frequently arising questions (FAQ) containing 10 questions and their answers which will be used as a dataset. By using a split ratio of 0.8 and a total of 60 questions, the resulting accuracy value is 93.33% and the error value is 6.66%.*

**Key Words :** Algorithm, application, chatbot, faq, naive bayes classifier

## 1. PENDAHULUAN

Larangan untuk bekerja di kantor di masa pandemi ini memaksa tidak sedikit dari perusahaan untuk mengambil kebijakan melakukan aktivitas bekerja di rumah atau biasa disebut *work from home* (WFH)

untuk para karyawannya, yang menyebabkan terbatasnya interaksi tatap muka. Hal ini sangat berpengaruh dalam pekerjaan terutama bagi para pegawai baru yang biasanya memiliki banyak pertanyaan yang ditanyakan pada rekan kerja seputar

pekerjaan yang baru dihadapi. Minimnya akses yang dimiliki dan sumber informasi yang berbeda beda untuk mendapatkan informasi tertentu membuat para pegawai baru kesulitan dalam bekerja padahal pertanyaan yang ditanyakan adalah pertanyaan yang sama dengan yang ditanyakan oleh pegawai satu dan lainnya.

Misalnya, para pegawai baru di GrabAds bingung bagaimana cara membuat laporan mingguan, membuat laporan akhir, implementasi *pixel*, penyelesaian masalah dan lain-lain. Pertanyaan tersebut ditanyakan oleh orang yang berbeda kepada orang yang sama tetapi jangka waktunya berbeda sehingga menyebabkan terbuangnya waktu karna menjawab pertanyaan yang sama berkali-kali. Dalam penelitian ini dikembangkan aplikasi *Chatbot* menggunakan algoritma *Naive Bayes Classifier* yang diharapkan dapat menjawab pertanyaan yang sering timbul terkait pekerjaan di GrabAds.

Tujuan utama dari *Chatbot* adalah untuk berkomunikasi dengan manusia dan dengan bantuan itu membuat banyak pekerjaan yang berlebihan menjadi lebih mudah bagi manusia. *Chatbot* tidak dibatasi waktu dan karena itu dapat melayani pengguna kapan saja. *Chatbot* menyediakan layanan pelanggan yang efisien dan dengan demikian bermanfaat bagi bisnis dan perusahaan. Oleh karena itu *Chatbot* dapat meningkatkan efisiensi [1].

Dalam penelitiannya Siddhi, Suyasha, Pranali, Manasi dan Anadkumar memperkenalkan beberapa teknik desain yang sudah banyak digunakan dalam pembuatan *Chatbot* diantaranya yaitu *sequence to sequence model* (seq2seq), *pattern matching*, *Long Short Term Memory* (LSTM), *Hybrid Emotion Interference Model* (HEIM), *Naive Bayes*, dan *Natural Language Processing* (NLP) untuk mendapatkan gambaran tentang bagaimana *Chatbot* dikembangkan serta membahas tentang dua hasil penelitian

*Chatbot* terdahulu yang menonjol yaitu ELIZA dan ALICE [1].

ALICE dan ELIZE sama-sama menggunakan teknik *pattern matching* dan representasi pengetahuan namun ALICE menggunakan algoritma pencocokan pola sederhana dan juga pola *template* yang mudah untuk representasi masukan dan keluaran. Algoritma pencocokan pola yang digunakan oleh ALICE mudah diimplementasikan dan bergantung pada pencarian pertama yang mendalam [1].

Penelitian mengenai algoritma *Naive Bayes Classifier* untuk tujuan yang berbeda telah banyak dilakukan sebelumnya, diantaranya adalah untuk mendeteksi *malware* yang dilakukan oleh Inda, Yesi dan Firdaus dimana algoritma *Naive Bayes Classifier* dipadukan dengan teknik *binning* (diskritasi) sehingga menghasilkan tingkat akurasi yang lebih baik [2], untuk klasifikasi dokumen konten *e-government* yang dilakukan oleh Akhmad dan Heru dengan akurasi keseluruhan klasifikasi sebesar 85 % [3].

Begitupula dengan penelitian mengenai *Chatbot* juga sudah pernah dilakukan sebelumnya, *Super Agent Chatbot "3S"* milik Herwin dan Khusaeri dengan tingkat akurasi 70 % berhasil dirancang sebagai media informasi [4]. Kemudian ada penelitian *Chatbot* untuk klasifikasi *intent* yang dilakukan oleh Muhammad Yusril, Rolly dan Safif dengan tingkat akurasi 63,63 % [5].

Moehammad Sarosa, Mochammad Junus dan Mariana dalam penelitiannya menggunakan algoritma *Naive Bayes* dan *Phase Reinforcement* untuk membuat *Chatbot* yang mampu memberikan latihan wawancara dalam bahasa inggris khususnya pada wawancara kerja yang bertujuan sebagai solusi alternatif untuk para pelajar yang memiliki kemampuan yang kurang dalam melakukan wawancara kerja dalam bahasa inggris. Penelitian ini

menggunakan algoritma *Naive Bayes* untuk mengklasifikasi hasil dari sebuah sesi wawancara kerja antara pengguna dan *interviewer-bot*. Terdapat tiga kategori dari percakapan, yaitu minta, potensi dan bakat. Klasifikasi kategori dihasilkan dari kalkulasi kemungkinan menggunakan Teorema Bayes. Penelitian tersebut menghasilkan perangkat lunak yang memiliki tingkat akurasi 86.93% [6].

Merry, Mohammad Syafrullah dan Hilman juga menggunakan algoritma *Naive Bayes* untuk mengembangkan aplikasi *Chatbot* yang mampu mencari informasi yang dibutuhkan mengenai gangguan pendengaran dengan menggabungkan algoritma tersebut dengan *Natural Language Processing (NLP)*. NLP digunakan untuk membantu *Chatbot* dalam menganalisa dan memahami bahasa alami manusia sementara algoritma *Naive Bayes* digunakan pada tahap pengklasifikasian teks yang dilatih sehingga mampu menangani permintaan pengguna dalam hal informasi tentang gangguan pendengaran secara akurat. Penelitian tersebut menghasilkan *Chatbot* yang memiliki akurasi sebesar 88.75% [7].

Darius dan Sophie dalam penelitiannya tentang *Chatbot* menyebutkan bahwa di masa depan, *Chatbot* dapat melengkapi atau bahkan menggantikan informasi tradisional, komunikasi, dan saluran penjualan seperti buletin, situs web, meja penjualan, atau *hotline* karna bisa digunakan kapanpun dan dimanapun, yang mana tidak mungkin dilakukan dengan saluran tradisional seperti situs web, buletin, atau *hotline* [8].

Dari hasil studi literatur yang sudah dilakukan tersebut serta mengacu pada keberhasilan penelitian sebelumnya maka peneliti menjadikannya sebagai referensi dalam mengembangkan aplikasi *Chatbot* menggunakan algoritma *Naive Bayes Classifier*.

## 2. METODE PENELITIAN

### Studi Literatur

Sebelum melakukan penelitian, penulis mempelajari dan membaca penelitian yang sudah dilakukan sebelumnya dalam bentuk jurnal ilmiah sebagai sumber informasi sekaligus referensi. Penulis juga membaca buku mengenai teori-teori yang berkaitan dengan penelitian. Hal tersebut penulis jadikan fondasi dan dasar acuan dalam penelitian yang akan dilakukan.

### Teknik Pengumpulan Data

Data yang digunakan dalam penelitian ini adalah kumpulan pertanyaan yang sering timbul (FAQ) yang didapatkan dari grup *Slack* yang digunakan perusahaan. Kumpulan pertanyaan yang sering timbul tersebut berbahasa inggris yang akan diterjemahkan kedalam bahasa indonesia untuk mempermudah penelitian. Data ini selanjutnya akan disebut sebagai *dataset*.

### Data Pre-processing

*Data Pre-processing* atau pra-pemrosesan data adalah proses dimana data dalam bentuk teks akan diolah sehingga dapat dimengerti oleh komputer. Teknik *data pre-processing* yang digunakan dalam penelitian ini adalah *tokenization*, *stopwords*, *lemmatization*, *stemming* dan *bag of words*.

### *Naive Bayes Classifier (NBC)*

Algoritma *Naive Bayes Classifier* dikembangkan berdasarkan dari teorema Bayes dan termasuk kedalam kategori algoritma *supervised learning*. Algoritma ini menggunakan *training dataset* (kumpulan data untuk dilatih) yang akan memandu komputer dalam menghasilkan keluaran yang sesuai dengan harapan.

Teorema Bayes adalah sebuah teorema yang terdiri dari dua penafsiran yang berbeda. Teorema Bayes menyatakan bagaimana seharusnya tingkat kepercayaan subjektif dapat berubah secara wajar ketika diberikan petunjuk baru. Teorema ini berasal dari penerapan teori probabilitas,

yaitu bagaimana mengetahui probabilitas dari dua penafsiran yang berbeda [9].

Algoritma *Naive Bayes Classifier* banyak digunakan karena memiliki komputasi yang lebih sederhana dalam implementasinya dibanding metode klasifikasi lainnya, meskipun mungkin dalam hal akurasi masih lebih baik jika digunakan metode C4.5, *K-Nearest Neighbors*, dan *Backpropagation Neural Network* [10].

Teorema Bayes dirumuskan dengan :

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \quad (1)$$

Dimana:

$P(A|B)$  = Kemungkinan dari kejadian *A* apabila terjadi kejadian *B* (*event A given event B*) atau disebut *posterior probability*.

$P(B|A)$  = Kemungkinan dari kejadian *B* apabila terjadi kejadian *A* (*event B given event A*) atau disebut *likelihood*.

$P(B)$  = Probabilitas atau kemungkinan (*B*) atau disebut *prior probability*. Berlaku ketentuan dimana  $P(B) \neq 0$ .

Langkah-langkah dalam proses *Naive Bayes Classifier* adalah sebagai berikut:

- Hitung jumlah dari kelas (*class*) atau label.
- Menghitung jumlah dari kasus per kelas (*class*).
- Kalikan semua variabel dari kelas (*class*).
- Lakukan komparasi dari hasil per kelas (*class*).

Contohnya, pada kalimat “Halo selamat pagi” maka kalimat tersebut akan dipecah menjadi ‘Halo’, ‘selamat’, ‘pagi’. Kemudian diberikan data pelatihan yang nantinya akan disimpan dalam bentuk daftar atau kamus yang memiliki kelas (*class*) dan kalimat sebagai atribut. Misalnya terdapat data latihan sebagai berikut :

*Class* : Salam

‘Salam sehat’

‘Salam sejahtera’

‘Halo selamat pagi’

*Class* : Laporan

‘Bagaimana cara mengerjakan laporan harian?’

‘Bagaimana cara mengerjakan laporan mingguan?’

‘Bagaimana cara mengerjakan laporan bulanan?’

Masukan pengguna: ‘Halo selamat pagi’

Istilah: Halo (cocok dengan *class* salam)

Istilah: selamat (cocok dengan *class* salam)

Istilah: pagi (cocok dengan *class* salam)

Maka dihasilkan klasifikasi Salam dan skor 3.

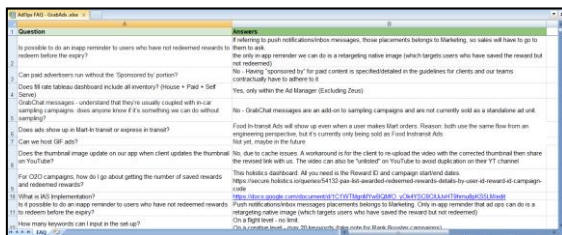
*Naive Bayes Classifier* dapat digunakan untuk menyelesaikan permasalahan klasifikasi melalui pendekatan probabilitas dengan asumsi prediktor variabel saling independen satu sama lain. Sehingga hasil dari model hanya bergantung pada sekumpulan variabel independen dan tidak saling mempengaruhi [9].

Kelebihan dari algoritma ini adalah apabila ada kelas yang telah didefinisikan sebelumnya maka akan menjadi mudah untuk diprediksi. Namun apabila data yang diberikan sebagai masukan (*input*) bukan merupakan milik kelas yang telah ditentukan sebelumnya maka menjadi tidak mungkin untuk dapat memprediksi kalimat keluaran (*output*). Oleh karena itu, karena kelemahan ini lah algoritma *Naive Bayes Classifier* memiliki kegunaan yang terbatas [1].

### 3. HASIL DAN PEMBAHASAN

#### Dataset

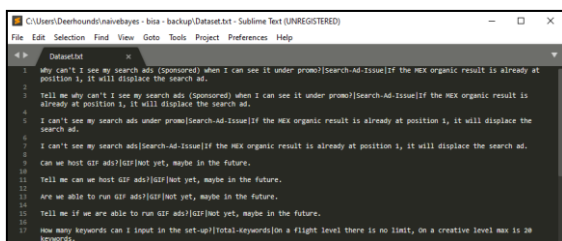
Dataset harus disortir dan dirapikan terlebih dahulu sebelum dapat digunakan yaitu dengan menghapus baris atau kolom yang kosong, menghapus symbol yang tidak memiliki arti, dan lain-lain.



Gambar 1. Dataset yang belum diolah

Gambar 1 merupakan *dataset frequently asked questions* (FAQ) sebelum diolah dalam format .xlsx. Data ini berisi 56 pertanyaan yang nantinya akan diolah dan dipilih 10 pertanyaan saja.

Dalam pengolahan *dataset* dibagi kedalam 3 bagian dan dipisahkan dengan tanda *vertical bar* (“|”) yaitu ; 1. Masukan pengguna (*input*), 2. *Category/class*, 3. Jawaban dari *bot* (*output*). Hasil dari data tersebut akan disimpan dalam format teks (.txt).



Gambar 2. Dataset yang sudah diolah

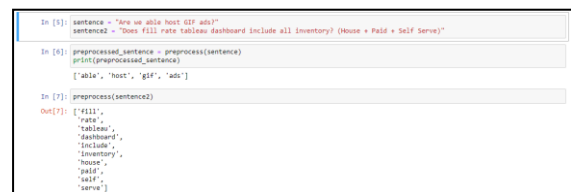
Gambar 2 merupakan *dataset frequently asked questions* (FAQ) yang sudah diolah sehingga dapat diproses dengan algoritma *Naive Bayes Classifier*.

#### Data Pre-processing

*Text Pre-processing* merupakan cara pemecahan teks ke dalam kata, yang mana kata tersebut akan menjadi kata kunci. Serangkaian langkah yang masuk ke dalam *preprocessing* diantaranya adalah *stemming*, *stopword removal* dan *tokenisasi* [11].

Sebelum *dataset* dapat diproses dengan algoritma *Naive Bayes Classifier*, *dataset* akan melalui tahap *pre-processing* (pra-pemrosesan) dimana tahap ini kalimat pertanyaan yang ada pada *dataset* akan melalui 2 proses yaitu *tokenization* (tokenisasi) dan penghapusan *stopwords*.

Pada metode *tokenization* terdapat 2 proses yang harus dilakukan yaitu merubah semua huruf besar menjadi kecil (*text to lowercase*) lalu membagi teks menjadi kumpulan kata tanpa memperhatikan keterhubungan diantara kata satu dengan yang lain serta peran dan posisinya pada kalimat, karakter diterima dalam kumpulan kata menurut abjad. Sementara pada metode *stopwords* jika kata pada *dataset* terdapat pada *stop word list* maka kata akan dihilangkan. Tetapi jika tidak terdapat di dalamnya maka proses akan berlanjut tanpa menghilangkan kata [3].



Gambar 3. Proses *Tokenization* dan *Stopwords*

Gambar 3 merupakan contoh data dalam bentuk kalimat setelah dilakukan proses *tokenization* dan *stopwords*, contoh kalimatnya adalah “Are we able host GIF ads?” dan “Does fill rate tableau dashboard include all inventory?(House + Paid + Self Serve)”.

Kemudian tiap kata akan dilakukan *tagging* agar dapat diekstrak kata kerja dan kata benda saja. Selanjutnya kata-kata yang ada didalam kalimat akan melalui proses *lemmatization* dan *stemming*. Proses ini bertujuan untuk menghapus kata agar dapat mengetahui kata asalnya (*stemming*) dan lalu akan dikelompokkan untuk menjadi satu item (*lemmatization*).

Pada metode *stemming* terjadi proses pemisahan antar imbuhan baik awalan,

akhirannya dengan kata dasar. Seluruh kata-kata yang akan dihilangkan telah disimpan pada ruang *stem list*, lalu data akan dicocokkan dengan *stem list*. Data yang sama dengan *stem list* akan dianggap *noisy* dan hilang [12].

```

In [14]: sentence
Out[14]: 'Are we able host gif ads?'

In [15]: words = extract_feature(sentence)
print(words)
['able', 'host', 'gif', 'ad']

In [17]: sentence2
Out[17]: 'Does fill rate tableau dashboard include all inventory? (House + Paid + Self Serve)'

In [18]: words = extract_feature(sentence2)
print(words)
['fill', 'rate', 'tableau', 'dashboard', 'include', 'inventory', 'house', 'self']
    
```

**Gambar 4. Proses Lemmatization dan Stemming**

Gambar 4 merupakan contoh data dalam bentuk kalimat setelah dilakukan proses *lemmatization* dan *stemming*. Hasilnya diperoleh dengan meng-ekstrak fitur dari sebuah kalimat.

Setelah itu, data akan dilakukan proses *bag of words*. Secara sederhana, ini adalah kumpulan kata untuk mewakili sebuah kalimat, yang mana akan mengabaikan urutan kemunculannya.

**Training Data dan Testing Data**

Data dibagi menjadi set pelatihan dan set pengujian. Proses ini dilakukan berulang kali untuk mendapatkan hasil pelatihan dan pengujian yang paling tinggi. Berikut adalah hasil dari pelatihan dan pengujian data dengan rasio perpecahan yang berbeda.

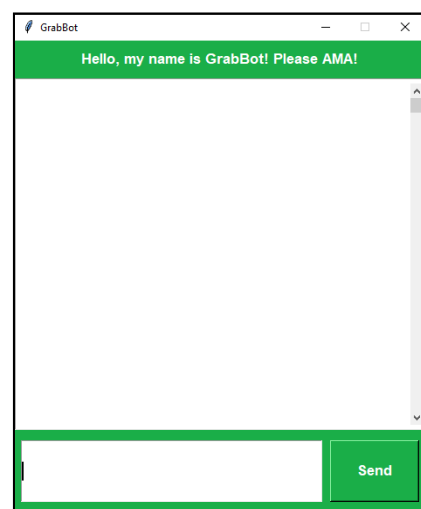
**Tabel 1. Hasil Training Set dan Test Set**

Split Ratio	Training Set Accuracy	Test Set Accuracy	Most Informative Features
0.5	1.0	0.9	82
0.6	1.0	0.75	90
0.7	1.0	0.83	94
0.8	1.0	1.0	100
0.9	1.0	1.0	100

Tabel 1 menunjukkan hasil dari *training set* dan *test set* dengan *split ratio* 0.5 sampai 0.9. Berdasarkan dari tabel diatas maka dapat disimpulkan bahwa semakin tinggi *split ratio* maka *training set* dan *test set* nya akan semakin akurat.

**Antarmuka Aplikasi Chatbot**

Antarmuka Aplikasi *Chatbot* dirancang menggunakan Tkinter. Tkinter adalah alat yang digunakan untuk menghubungkan grafis pada bahasa pemrograman Python. Tkinter merupakan salah satu *library* yang paling sering digunakan untuk mengembangkan atau merancang GUI (*Graphical User Interface*). Dengan Tkinter ini penulis berhasil merancang antarmuka sesuai dengan rancangan aplikasi *chatbot* yang diinginkan.

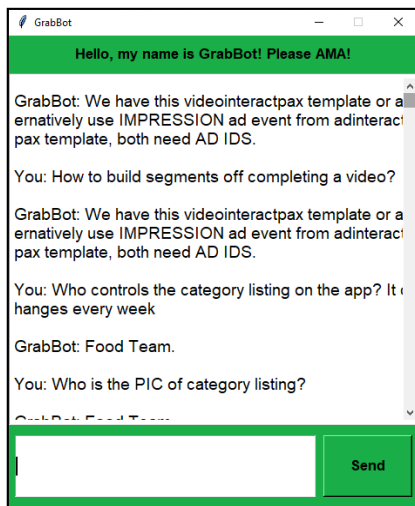


**Gambar 5. Antarmuka Aplikasi Chatbot**

Gambar 5 merupakan tampilan antarmuka dari aplikasi *Chatbot* yang berhasil dirancang berdasarkan rancangan yang diinginkan.

**Pengujian Sistem**

Berdasarkan hasil dari *training set* dan *test set* maka dalam penelitian ini digunakan *split ratio* sebesar 0,8 untuk kemudian dilakukan pengujian dengan memberikan pertanyaan kepada *Chatbot* melalui GUI (*Graphical User Interface*) yang sebelumnya telah berhasil dibuat sebanyak 60 pertanyaan. Pertanyaan yang diajukan akan berbeda-beda namun memiliki konteks yang sama.



Gambar 6. Hasil Pengujian Sistem

Gambar 6 merupakan contoh dari pertanyaan yang diajukan kepada *Chatbot*. Dari total 60 pertanyaan yang diajukan, *Chatbot* berhasil menjawab sebanyak 56 pertanyaan dengan benar dan jawaban yang salah sebanyak 4 pertanyaan.

Untuk dapat mengetahui nilai akurasi dan kesalahan dari jawaban *Chatbot* maka digunakan rumus sebagai mana yang dijelaskan pada persamaan (2) dan (3) :

$$\text{Akurasi} = \frac{\text{Total pengujian yang benar}}{\text{Total pengujian}} \times 100 \% \quad (2)$$

$$\text{Kesalahan} = \frac{\text{Total pengujian yang salah}}{\text{Total pengujian}} \times 100 \% \quad (3)$$

Dengan menggunakan rumus tersebut maka nilai akurasi dan kesalahannya adalah :

$$\text{Akurasi} = \frac{56}{60} \times 100 \% = 93,33 \% \quad (4)$$

$$\text{Kesalahan} = \frac{4}{60} \times 100 \% = 6,66 \% \quad (5)$$

Dengan menggunakan *split ratio* sebesar 0,8 dan total 60 pertanyaan maka dapat dilihat pada persamaan (4) dan (5) bahwa dihasilkan nilai akurasi sebesar 93,33 % dan nilai kesalahan sebesar 6,66 %.

#### 4. SIMPULAN

Salah satu kelebihan dari Algoritma *Naive Bayes Classifier* dalam pengembangan *Chatbot* adalah kemudahannya dalam hal komputasi maupun implementasi dan itu

sudah dibuktikan dalam penelitian ini kebenarannya. Setelah melakukan penelitian ini penulis menyimpulkan bahwa :

1) Penulis berhasil mengumpulkan pertanyaan yang sering timbul (FAQ) kedalam satu sumber untuk digunakan dalam pengembangan *Chatbot*.

2) Penerapan algoritma *Naive Bayes Classifier* memungkinkan *Chatbot* untuk dapat menjawab pertanyaan yang diberikan oleh pengguna seperti yang ditunjukkan pada pengujian sistem.

3) Terdapat beberapa metode yang dapat digunakan untuk meningkatkan akurasi dari jawaban seperti metode *tokenization*, *stop words*, *lemmatization*, *stemming* dan *bag of words*.

4) *Chatbot* tidak dapat mengenali persamaan kata dari pertanyaan yang diajukan sehingga mengakibatkan *Chatbot* memberikan jawaban yang salah.

Dengan menggunakan *split ratio* sebesar 0,8 dan total 60 pertanyaan maka dihasilkan nilai akurasi sebesar 93,33 % dan nilai kesalahan sebesar 6,66 %.

#### DAFTAR PUSTAKA

- [1] S. Pardeshi, S. Ovhal, P. Shinde, M. Bansode, and A. Birajdar, "A Survey on Consensus Algorithms used in Blockchain Platforms," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 5, pp. 9155–9162, 2020, doi: 10.30534/ijatcse/2020/323952020.
- [2] I. Anggraini and Y. N. Kunang, "Telematika Penerapan Naive Bayes pada Pendeteksian Malware dengan Diskritisasi Variabel," vol. 13, no. 1, pp. 11–21, 2020.
- [3] A. P. Wijaya and H. A. Santoso, "Naive Bayes Classification pada Klasifikasi Dokumen Untuk Identifikasi Konten E-Government," *J. Appl. Intell. Syst.*, vol. 1, no. 1, pp.

- 48–55, 2016, [Online]. Available: <https://publikasi.dinus.ac.id/index.php/jais/issue/view/76>.
- [4] H. H. Herwin, “Super Agent Chatbot ‘3S’ Sebagai Media Informasi Menggunakan Metoda Natural Language Processing(NLP),” *J. Teknol. Dan Open Source*, vol. 2, no. 1, pp. 53–64, 2019, doi: 10.36378/jtos.v2i1.144.
- [5] M. Y. H. Setyawan, R. M. Awangga, and S. R. Efendi, “Comparison Of Multinomial Naive Bayes Algorithm And Logistic Regression For Intent Classification In Chatbot,” *Proc. 2018 Int. Conf. Appl. Eng. ICAE 2018*, no. October, pp. 1–5, 2018, doi: 10.1109/INCAE.2018.8579372.
- [6] M. Sarosa, M. Junus, M. U. Hoesny, Z. Sari, and M. Fatnuriyah, “Classification technique of interviewer-bot result using naive bayes and phrase reinforcement algorithms,” *Int. J. Emerg. Technol. Learn.*, vol. 13, no. 2, pp. 33–47, 2018, doi: 10.3991/ijet.v13i02.7173.
- [7] M. Anggraeni, M. Syafrullah, and H. A. Damanik, “Literation Hearing Impairment (I-Chat Bot): Natural Language Processing (NLP) and Naive Bayes Method,” *J. Phys. Conf. Ser.*, vol. 1201, no. 1, 2019, doi: 10.1088/1742-6596/1201/1/012057.
- [8] D. Zumstein and S. Hundertmark, “Chatbots : an interactive technology for personalized communication and transaction,” *IADIS Int. J. www/Internet*, vol. 15, no. 1, pp. 96–109, 2018.
- [9] R. Primartha, *Algoritma Machine Learning*. Informatika Bandung, 2021.
- [10] Z. Sari, M. Sarosa, and S. Suhari, “‘Si Tole’ Chatterbot untuk Melatih Rasa Percaya Diri Menggunakan Naive Bayes Classification,” *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 7, no. 1, pp. 64–71, 2018, doi: 10.22146/jnteti.v7i1.402.
- [11] F. Fathoni, E. Afrianti, and R. I. Heroza, “Klasifikasi Teks dengan Naive Bayes Classifier (NBC) untuk Pengelompokan Keterangan Laporan dan Durasi Recovery Time Laporan Gangguan Listrik PT. PLN (Persero) WS2JB Area Palembang,” *JSI J. Sist. Inf.*, vol. 12, no. 1, pp. 1955–1961, 2020, doi: 10.36706/jsi.v12i1.9586.
- [12] F. Handayani and S. Pribadi, “Implementasi Algoritma Naive Bayes Classifier dalam Pengklasifikasian Teks Otomatis Pengaduan dan Pelaporan Masyarakat melalui Layanan Call Center 110,” *J. Tek. Elektro*, vol. 7, no. 1, pp. 19–24, 2015, doi: 10.15294/jte.v7i1.8585.