Clean Energy and Smart Technology
Volume 03, Number 01, October 2024
DOI : https://doi.org/10.58641/

e-ISSN : 2964-2647

IMPLEMENTATION OF CHATBOT AS A VIRTUAL ASSISTANT IN PUBLIC SERVICES AND ASSISTANCE IN THE SOCIAL SERVICES OF WONOSOBO DISTRICT USING THE TF-IDF METHOD

Sagita Efendi[1)], Hidayatus Sibyan[2) *], Nur Hasanah[3)]
[1)2)3)] Universitas Sains Al-Qur'an, Indonesia
[1)]sagitaefendi@gmail.com, [2)]hsibyan@unsiq.ac.id, [3)]nurhasanah@unsiq.ac.id

*hsibyan@unsiq.ac.id

Submitted : 3 October 2024 | **Accepted** : 26 October 2024 | **Published** : 31 October 2024

Abstract: Information needs are important in public service to give users an initial understanding of the services, the lack of information can make the public service doesn't work like its function. Based on that reason, a study has been done to create a system that can provides information and assistance on Dinas Sosial PMD at Wonosobo Regency. The study creates chatbot named HADIR bot as a virtual assistant on public services and assistance in Dinas Sosial PMD at Wonosobo regency with TF-IDF method. The goals of this chatbot are to support public service and also virtual assistant that can run 24 hours a day based on the users request or question in chatbot, this chatbot also provides initial description and frequently asked question that often asked to the employees, and provides basic information and document template about list of the assistance. The design of the chatbot uses UML (Unified Modelling Language) like Use Case, Activity Diagram, and Sequence Diagram. This Chatbot created with telegrams features named BotFather, with the function made by Python and text editor uses Visual Studio Code, the chatbot runs in Telegram with TF-IDF method in term weighting and cosine similarity method to evaluate the similarities between document.

Keywords: Chatbot, Telegram, Public Service, TF-IDF, Cosine Similarity

1. INTRODUCTION

The uneven distribution of information received and accessed by the public regarding assistance programs, repeated questions regarding templates or formats for completeness of assistance documents, the Telegram platform not yet available in public services at the PMD social service, and limited working hours are the basis for creating a Telegram chatbot that can function as service information and assistance. In organizing public services, organizers are required to apply the principles of effectiveness, efficiency, innovation and quality commitment. Because the orientation of public services is public satisfaction, the public gets services according to what is expected or even exceeds public expectations.

The Virtual Assistant that will be created is a ChatBot on the Telegram application platform which in principle is an artificial intelligence that can understand questions from the public/service recipients and can provide answers automatically. In the development of the digital era like today, chatbots are still one of the technological innovations that are widely used as a medium of communication with users in various fields, including public services. (Mustaqim et al., 2023).

So far, in the practice of public services at the PMD Social Service, there has been no service platform through the Telegram application. Limited service hours are also one of the obstacles to public services at the PMD Social Service. Virtual Assistants that can serve the public 24 hours a day can be a solution to limited service hours, can provide information on certain documents and document formats, and can clarify the role of the PMD Social Service as a facilitator and not as a decision maker on assistance menus for certain groups for the general public to the Village Government apparatus. Of course, this service is not a service that can solve problems technically, but it is hoped that with this virtual assistant, it can provide fast, precise, direct, and automatic answers for users. Natural Language Processing or NLP studies communication between humans and computers through natural language. Chatbot technology is one form of Natural Language Processing application, which is one of the fields of artificial intelligence that studies communication between humans and computers through natural language. (Mulyatun et al., 2021). The form of communication that occurs between humans and computers is through conversation using written media. The creation of this chatbot is based on applying the TF-IDF method to simplify questions entered by users and then processed into basic words, to be further processed using preprocessing into keywords with response output that has been labeled/tagged according to the question. TF-IDF or Term Frequency Inverse Document Frequency is a method used to determine the frequency value of a word in a document or article and also the frequency in many documents. Term frequency selection is carried out to get a few terms in order to reduce the value of less relevant occurrences. The following is the equation used to calculate term frequency (Sukma, Allfian, 2019).

The use of the Cosine Similarity method is generally together with the TF-IDF method and has the principle of finding similarities between two vectors based on the multiplication value of the cosine angles of both. (Lahitani et al., 2016). The cosine similarity method is one method that can be used to match the questions asked with the knowledge base owned by the chatbot with high accuracy and fast execution time (Kowsher et al., 2019). The use of this method is suitable for the "HADIR-Bot" chatbot as a virtual assistant because the questions are the same and repeated so that these words can be the basis for finding answers from the chatbot's knowledge library data. The hope is that each question can be answered with the cosine similarity method from the weighting results of the TF-IDF method.

The basis of the name of the chatbot to be created is HADIR-Bot, which is the Motto/Tagline of the Wonosobo Regency Social Service, Community Empowerment and Village Service, namely HADIR, which is a combination of Humanist, Adaptive, Dedicated, Inclusive, and Responsive.

From the existing problem formulation, a Telegram chatbot will be created as a virtual assistant to support public services at the PMD Social Service which can answer questions from service users and provide basic information and document templates or assistance document formats using the TF-IDF and Cosine Similarity methods and can be accessed 24 hours by service users.

2. METHOD

The type of research used in this study is quantitative research. Quantitative research methods are one type of research whose specifications are systematic, planned and clearly structured from the beginning to the creation of the research design. The type of data used is qualitative data because the data is in the form of words, schemes, and images. And the data used is internal data, the internal data used in this study is data in the form of Standard Operating Procedures (SOP) in the form of images, templates for assistance requirement documents, links to activity progress in each field, and the names of employees at Dinsos PMD along with the organizational structure.

The data analysis method that will be carried out by the author is the creation of a Telegram chatbot that will be created from Telegram's own features, namely through BotFather and developed through Python scripts as a Programming Language, Data Library which will be used as access to retrieve answer templates from users using JSON files that can display text and images related to Service Operational Standards (SOP). And

The data collection method used by the author is observation, literacy studies, and interviews with program admins so that internal data is obtained. Observations in this study were conducted directly because the author works at Dinsos PMD, so that observations can be made of the general public and other service users who come. In this case, in addition to the general public, there are also several village officials as service users. Literacy studies were conducted by the author in searching for theories, studying the advantages of previous research in accordance with the research conducted, and studying the completeness of assistance documents and technical instructions from the results of interviews. Interviews conducted by researchers were with employees or employees of Dinsos PMD Wonosobo Regency which produced data in the form of internal data.

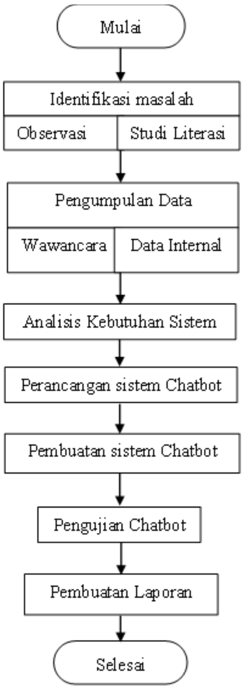


Figure 1 Chatbot Research Flow

The data processing method used by the author in this case uses Natural Language Processing which includes Text Preprocessing. The use of text preprocessing is to process a sentence or question from the user into a language that is easier for the chatbot to understand. The steps are Case Folding, Tokenizing, Stopword Removal, and Stemming which then produce keywords in the form of words per word which then from the results of this text preprocessing will be weighted using the TF-IDF (Term Frequency- Inverse Document Frequency) method and the Cosine Similarity method to find similarities between documents so that the chatbot can provide the appropriate answer if there is more than one answer that is similar in the arrangement of words.

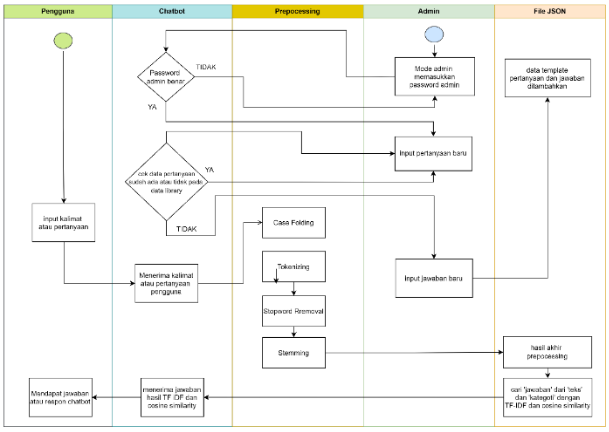

Pengguna
Chatbot
Prepocessing
Admin
File JSON
Password admin benar
TIDAK
YA
Mode admin memasukkan password admin
data template pertanyaan dan jawaban ditambahkan
Input pertanyaan baru
cek data pertanyaan sudah ada atau tidak pada data library
YA
TIDAK
input kalimat atau pertanyaan
Case Folding
Tokenizing
Menerima kalimat atau pertanyaan pengguna
Input jawaban baru
Stopword Rremoval
Stemming
hasil akhir prepocessing
Mandapat jawaban atau respon chatbot
menerima jawaban hasil TF IDF dan cosine similarity
cari 'jawaban' dari 'teks' dan 'kategori' dengan TF-IDF dan cosine similarity

Figure 2 Activity Diagram Chatbot

The following is an explanation of the activity diagram of the chatbot:
1. The user inputs a question sentence to the chatbot
2. The chatbot receives a question sentence from the user, then in this case the chatbot will check whether the question is in the data library or not. If yes, the chatbot will process the question sentence with the preprocessing method, if not, the chatbot will send a response in the form of a question that does not exist.
3. The user's question enters the preprocessing stage, and after the preprocessing results are found, the question sentence from the user is then weighed by the chatbot to weigh the weight of the words that appear from the preprocessing results and if there is more than one document that appears, the chatbot will rank the documents. The document with the highest ranking result (highest score) is the one that will be displayed to the user.
4. Here it is also explained for the activity diagram of the Admin, the admin can add to the JSON file in the form of a question and answer template, by entering the command "/mode_admin" then inputting the admin password, if the password is correct it will enter the stage of entering a new question, if the password is wrong it will return to the response enter the admin password.
5. The stage of successfully logging in or entering the admin password correctly, the chatbot will send a response to enter a new example question, after the admin enters a new question the chatbot will re-check whether the question is the same or not with the question in the data library, if the same will be asked to enter another question. And if not the same will be followed by a response to enter the answer to the question that was entered earlier.

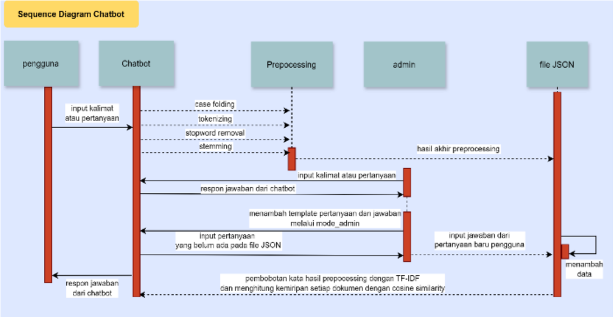

Sequence Diagram Chatbot
pengguna
Chatbot
Prepocessing
admin
file JSON
input kalimat atau pertanyaan
case folding
tokenizing
stopword removal
stemming
hasil akhir preprocessing
input kalimat atau pertanyaan
respon jawaban dari chatbot
menambah template pertanyaan dan jawaban melalui mode_admin
input pertanyaan yang belum ada pada file JSON
input jawaban dari pertanyaan baru pengguna
menambah data
respon jawaban dari chatbot
pembobotan kata hasil prepocessing dengan TF-IDF dan menghitung kemiripan setiap dokumen dengan cosine similarity

Figure 3 Chatbot Sequence Diagram

The method that the author uses for the system development method is by adding library data in the JSON file from admin mode via chatbot. In this mode, the admin can add knowledge from the chatbot through inputting question and answer templates.

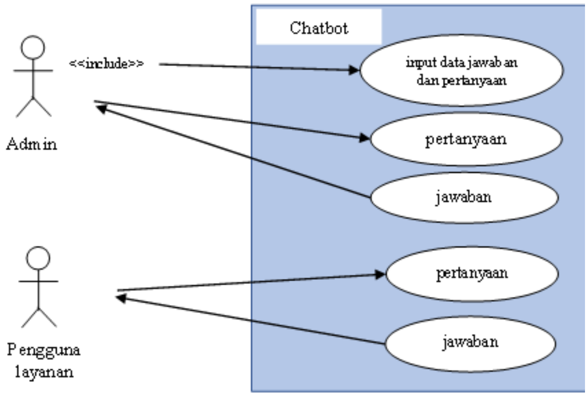


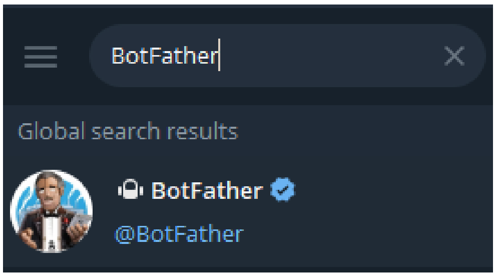
Figure 4 Chatbot System Use Case

Service users and admins have different access, Admins can input data using the admin mode method, by entering commands and passwords and can ask questions to the chatbot and get answers from the chatbot. While service users can only ask and get answers from the chatbot.

TF-IDF (Term Frequency-Inverse Document Frequency) method as a weighting method for each chatbot answer. The chatbot question and answer template as the basis for searching for answers from keywords from text preprocessing results will be stored in the data library in the form of a JSON file, the question and answer templates that already exist in the data library are then grouped or subheading so that the search for answers becomes faster. The method is to group the question words from questions such as what, who, how, where, and how much. Where the principle of the TF-IDF method is to calculate the weight of a term or word, if the question subheading is done, the faster the chatbot can find answers from the data library. Before using this TF-IDF method, the author first groups the list of questions and answers according to the question words from the question. Examples of question words are “what”, “who”, “where”, “why”, “when”, and “how”.

3. RESULT AND DISCUSSION

This stage is a follow-up to the chatbot design stage, namely creating a chatbot with Telegram features via BotFather. The chatbot application created is a Telegram chatbot application as a virtual assistant in public services at the Dinsos PMD Wonosobo Regency. In this study, the author used an existing feature on Telegram called BotFather to create a bot. To create a bot through the BotFather feature, it can be accessed via Telegram by searching for the keyword "BotFather".



BotFather

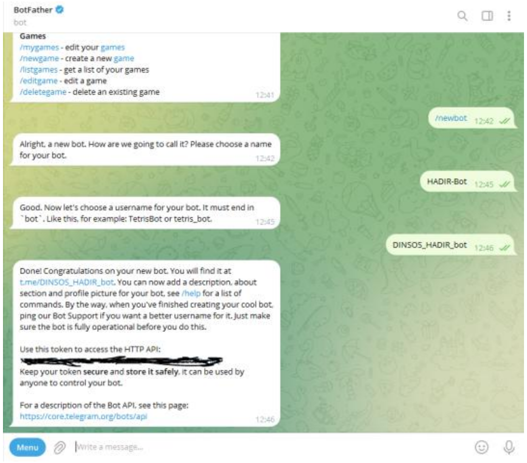
Global search results

BotFather

@BotFather

Figure 5 BotFather Search Results On Telegram

Figure 5 shows the BotFather search via Telegram which is used to create a bot. Click on the Start button to create a bot, by typing the command "/newbot", it must be remembered here it must be with a slash. If you have entered the command, it will go to the next command, namely giving the bot a name and username.



BotFather
bot

Games
/mygames - edit your games
/newgame - create a new game
/listgames - get a list of your games
/editgame - edit a game
/deletegame - delete an existing game
12:41

/newbot 12:42

Alright, a new bot. How are we going to call it? Please choose a name for your bot. 12:42

HADIR-Bot 12:45

Good. Now let's choose a username for your bot. It must end in `bot`. Like this, for example: TetrisBot or tetris_bot. 12:45

DINSOS_HADIR_bot 12:46

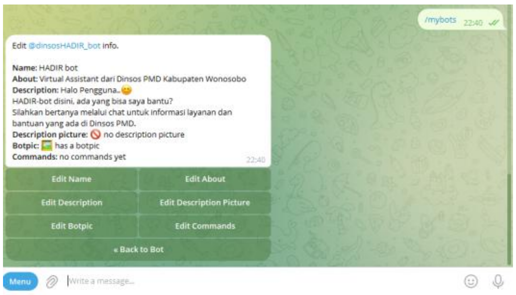
Done! Congratulations on your new bot. You will find it at t.me/DINSOS_HADIR_bot. You can now add a description, about section and profile picture for your bot, see /help for a list of commands. By the way, when you've finished creating your cool bot, ping our Bot Support if you want a better username for it. Just make sure the bot is fully operational before you do this.

Use this token to access the HTTP API:
[redacted]
Keep your token **secure** and **store it safely**, it can be used by anyone to control your bot.

For a description of the Bot API, see this page:
https://core.telegram.org/bots/api
12:46

Figure 6 Naming and Username of Bot

After the chatbot is successfully created, there will be a notification in the form of a username and bot token as in Figure 6. This bot token must be stored as well as possible and should not be shared with others, because if this bot token is shared, it is feared that the bot can be hacked or cannot be accessed. After that, you can enter the bot token and chatbot username in the Python script to be able to connect the command in Python with the Telegram chatbot.



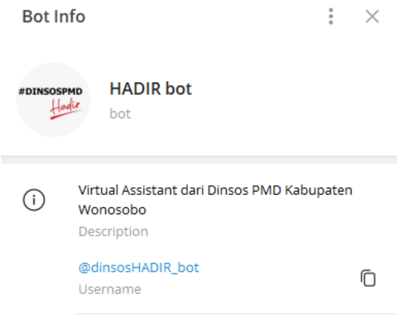
/mybots 22:40

Edit @dinsosHADIR_bot info.

Name: HADIR bot
About: Virtual Assistant dari Dinsos PMD Kabupaten Wonosobo
Description: Halo Pengguna..😊
HADIR-bot disini, ada yang bisa saya bantu?
Silahkan bertanya melalui chat untuk informasi layanan dan bantuan yang ada di Dinsos PMD.
Description picture: 🚫 no description picture
Botpic: 🖼 has a botpic
Commands: no commands yet
22:40

Figure 7 “/mybot” Command Display”

Editing bot name, bot description, bot profile photo, about using the command "/mybots" and select on the bot. Here the author only edits the important things as chatbot information, namely about, description, and description picture.



Bot Info

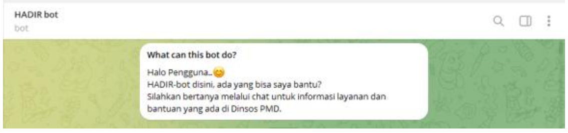
HADIR bot
bot

Virtual Assistant dari Dinsos PMD Kabupaten Wonosobo
Description

@dinsosHADIR_bot
Username

Figure 8 Bot Info View

After finishing editing the chatbot through BotFather, the next step is coding through Python tools. This coding is done to connect the Telegram chatbot with logical functions according to needs. In this chatbot, the author will only display coding in the form of preprocessing stage functions (case folding, tokenizing, stopword removal, and stemming) TF-IDF and cosine similarity methods, as well as JSON file library data that has been processed by subheading or tagging question types. and the results of coding through python as in the following image:



HADIR bot
bot

What can this bot do?

Halo Pengguna..😊
HADIR-bot disini, ada yang bisa saya bantu?
Silahkan bertanya melalui chat untuk informasi layanan dan bantuan yang ada di Dinsos PMD.

Figure 9 Initial View of HADIR Bot

This chatbot is designed and built through several stages, the first stage is the input of questions that will be given by the user. After the user asks the chatbot a question, the sentence will be processed to the preprocessing stage. The input sentences used in this chatbot use EYD and non-standard or informal language. So that every sentence received by the chatbot will be further processed to obtain data processing results in the form of optimal, maximum and simple answers so that they are easily accepted by users who are mostly from the general public. This is also so that the response given by the chatbot is as expected. Next, text processing will be carried out, namely the Preprocessing stage. The initial stage of processing text that enters the chatbot is case folding, namely changing the input sentence to lowercase letters.

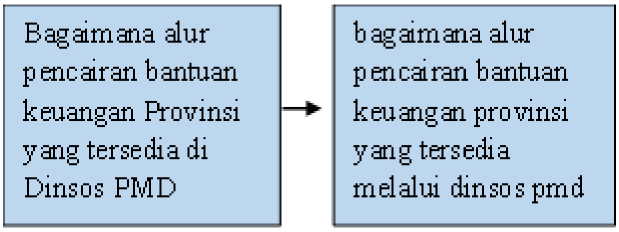
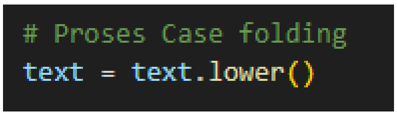

Bagaimana alur pencairan bantuan keuangan Provinsi yang tersedia di Dinsos PMD
bagaimana alur pencairan bantuan keuangan provinsi yang tersedia melalui dinsos pmd

Figure 10 Illustration of Folding Case



```
# Proses Case folding
text = text.lower()
```

Figure 11 Case Folding Script

The second stage of preprocessing is tokenizing or separating words in a sentence based on space characters from the previous processing results, namely case folding..

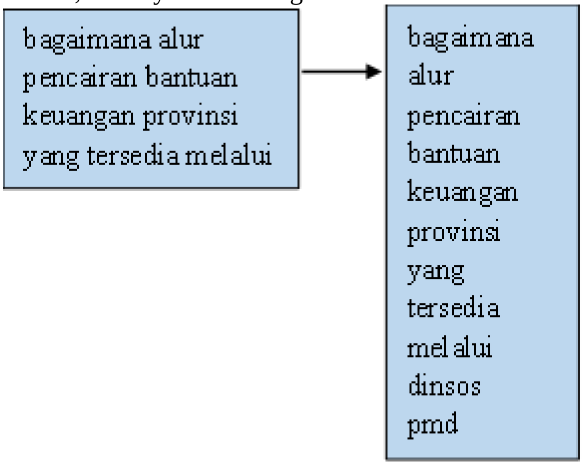
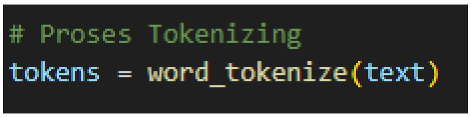


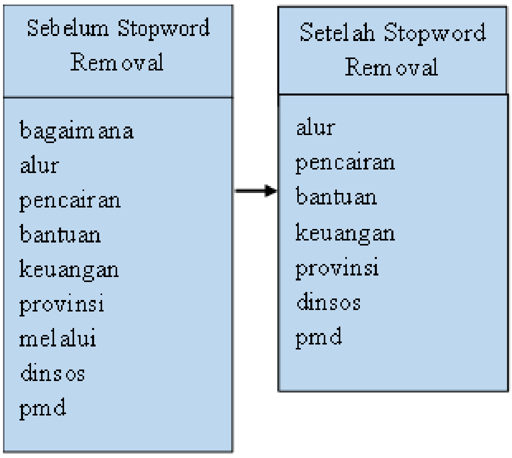
Figure 11 Illustration of Tokenizing



```
# Proses Tokenizing
tokens = word_tokenize(text)
```

Figure 12 Tokenizing Script

The third stage of preprocessing is stop word removal in this stage the words needed in the input of questions or sentences will be removed such as repeated words and conjunctions aiming to simplify word processing by the chatbot. At this stage the author uses the Sastrawi library which contains spoken and written language and has been arranged systematically.



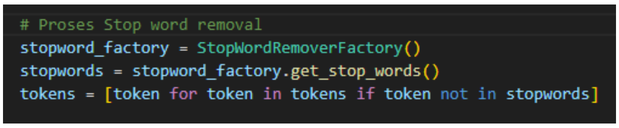
Sebelum Stopword Removal

bagaimana
alur
pencairan
bantuan
keuangan
provinsi
melalui
dinsos
pmd

Setelah Stopword Removal

alur
pencairan
bantuan
keuangan
provinsi
dinsos
pmd

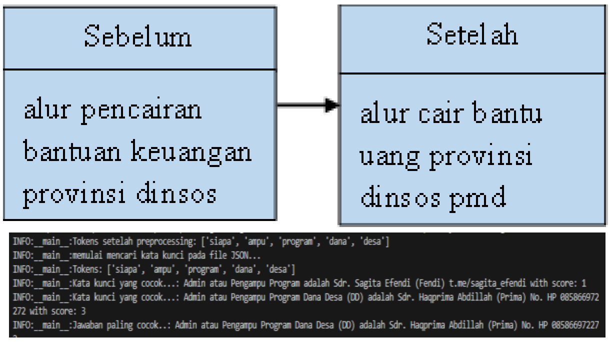
Figure 13 Illustration of Stopword Removal



```
# Proses Stop word removal
stopword_factory = StopWordRemoverFactory()
stopwords = stopword_factory.get_stop_words()
tokens = [token for token in tokens if token not in stopwords]
```

Figure 14 Stop Word Removal Script

The last stage of preprocessing is stemming, this stage is done to get the root word of a word by removing the affixes in the prefix, insertion, suffix and combination of affixes from a word. This process needs to be done as a step to standardize the words in the document into root words. Stemming on the Indonesian language data set is done several times, namely removing the affixes at the end of the word (for example "-lah", "-kah", "-tah" or "-pun"). Furthermore, this process is repeated again to remove inflectional suffixes (-ku, -mu, or -nya), then removing derivational suffixes, namely removing the final affix ("-i", "-an", or "-kan"), if the affix ends with the letter "-k" then the letter will be removed. If these affixes are not found, then the process of deleting the affixes is not carried out. After that, the process of deleting derivational prefix affixes is carried out, namely deleting affixes at the beginning of words (for example be-, di-, ke-, me-, pe-, se-, and te-). This process will stop if there are conditions such as a combination of initial and final affixes which is not allowed. Then there is the same prefix as the prefix that has been deleted. And then the 3 prefixes will be removed. The final process is if the processes that have been carried out do not provide a root word for the word, then the original word before stemming will be returned..



Sebelum

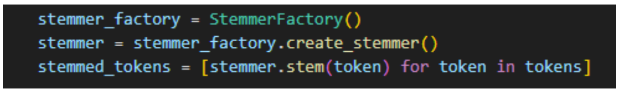
alur pencairan
bantuan keuangan
provinsi dinsos

Setelah

alur cair bantu
uang provinsi
dinsos pmd

```
INFO:__main__:Tokens setelah preprocessing: ['siapa', 'ampu', 'program', 'dana', 'desa']
INFO:__main__:memulai mencari kata kunci pada file JSON...
INFO:__main__:Tokens: ['siapa', 'ampu', 'program', 'dana', 'desa']
INFO:__main__:Kata kunci yang cocok...: Admin atau Pengampu Program adalah Sdr. Sagita Efendi (Fendi) t.me/sagita_efendi with score: 1
INFO:__main__:Kata kunci yang cocok...: Admin atau Pengampu Program Dana Desa (DD) adalah Sdr. Haqprima Abdillah (Prima) No. HP 085866972
272 with score: 3
INFO:__main__:Jawaban paling cocok..: Admin atau Pengampu Program Dana Desa (DD) adalah Sdr. Haqprima Abdillah (Prima) No. HP 08586697227
```

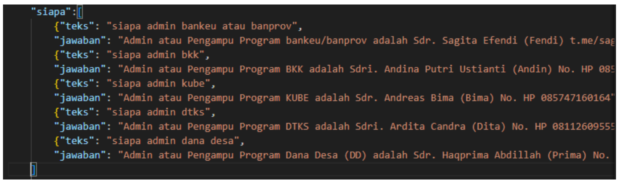
Figure 15 Stemming Illustration



```
stemmer_factory = StemmerFactory()
stemmer = stemmer_factory.create_stemmer()
stemmed_tokens = [stemmer.stem(token) for token in tokens]
```

Figure 16 Stemming Process Script

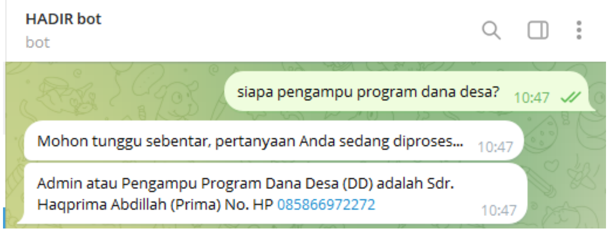
When the Preprocessing process is complete, the chatbot will enter the Training process stage. After the Training process stage is complete, a model will be produced that will be used in a process called Testing. At this testing stage, the chatbot will be tested to answer questions from service users. And in the testing stage, it will produce output from each user question in the form of questions from users and answers for users. This data will be an indicator to determine whether the chatbot's answer is in accordance or not as expected. The following is an example of a JSON file with the subheading "who" used in the chatbot.



```
"siapa":[
    {"teks": "siapa admin bankeu atau banprov",
    "jawaban": "Admin atau Pengampu Program bankeu/banprov adalah Sdr. Sagita Efendi (Fendi) t.me/sag
    {"teks": "siapa admin bkk",
    "jawaban": "Admin atau Pengampu Program BKK adalah Sdri. Andina Putri Ustianti (Andin) No. HP 085
    {"teks": "siapa admin kube",
    "jawaban": "Admin atau Pengampu Program KUBE adalah Sdr. Andreas Bima (Bima) No. HP 085747160164"
    {"teks": "siapa admin dtks",
    "jawaban": "Admin atau Pengampu Program DTKS adalah Sdri. Ardita Candra (Dita) No. HP 08112609555
    {"teks": "siapa admin dana desa",
    "jawaban": "Admin atau Pengampu Program Dana Desa (DD) adalah Sdr. Haqprima Abdillah (Prima) No.
]
```

Figure 17 Example of Training Data Subheading “Who”

Example of questions on the Telegram chatbot:



HADIR bot
bot

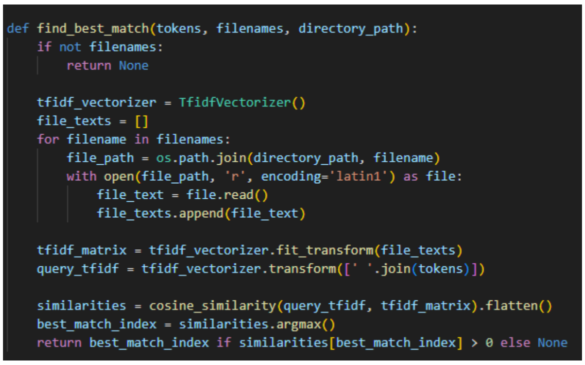
siapa pengampu program dana desa? 10:47

Mohon tunggu sebentar, pertanyaan Anda sedang diproses... 10:47

Admin atau Pengampu Program Dana Desa (DD) adalah Sdr. Haqprima Abdillah (Prima) No. HP 085866972272 10:47

Figure 18 Example of Chatbot Question “Who is the administrator of the village fund program?”

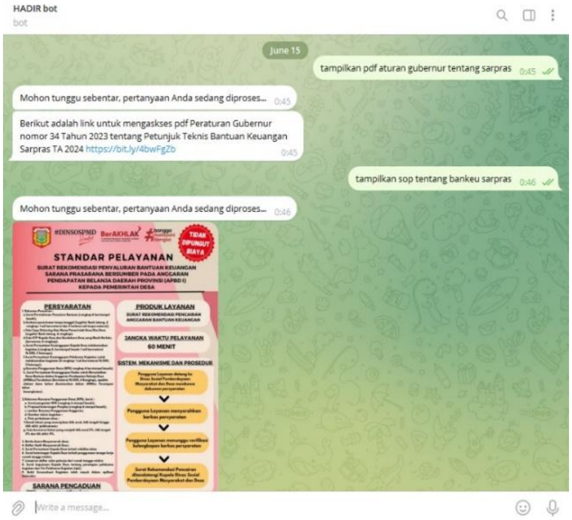
In the Python terminal, the process can be displayed starting from the preprocessing, then what documents match the question, then the weight of the document is also displayed, after finishing calculating the weight, the answer with the most weight will be displayed. The following script is written in Python for the TF-IDF and Cosine Similarity methods as follows:



```python
def find_best_match(tokens, filenames, directory_path):
    if not filenames:
        return None

    tfidf_vectorizer = TfidfVectorizer()
    file_texts = []
    for filename in filenames:
        file_path = os.path.join(directory_path, filename)
        with open(file_path, 'r', encoding='latin1') as file:
            file_text = file.read()
            file_texts.append(file_text)

    tfidf_matrix = tfidf_vectorizer.fit_transform(file_texts)
    query_tfidf = tfidf_vectorizer.transform([' '.join(tokens)])

    similarities = cosine_similarity(query_tfidf, tfidf_matrix).flatten()
    best_match_index = similarities.argmax()
    return best_match_index if similarities[best_match_index] > 0 else None
```

Figure 19 TF-IDF script and Cosine Similarity

In system testing, a file with the JSON extension (.json) is used as a data library with the name "bank_data.json" for a list of questions and answer templates in the form of text that will be processed by the chatbot from incoming user questions, using a file in the form of a photo with the JPG extension (.jpg) from the results of an interview with the Program Holder which will later be used as a means of information for questions about the service flow, the requirements needed and service times. In addition, the author also uses text in the form of links for pdf files. This chatbot is programmed with conditions or criteria if the user types the word "SOP" in the question to automatically display an answer only in the form of a service SOP at the PMD social service, if there is no word "SOP" then the chatbot will process questions in the form of text only from the user. If the user wants to display an answer in the form of a format or template for a document, rules or technical instructions about the assistance program, they can start the command with the keyword "pdf". Questions with the prefix "who" will display the name of the person and the contact that can be contacted, while questions with the word "what" will display an answer in the form of an understanding of the question keyword. The following is an example of user interaction with a chatbot with commands to display photo answers, answers in the form of definitions of keywords, names of people according to keywords, and answers in the form of PDF download links.



HADIR bot
bot

June 15

tampilkan pdf aturan gubernur tentang sarpras 0:45

Mohon tunggu sebentar, pertanyaan Anda sedang diproses... 0:45

Berikut adalah link untuk mengaskses pdf Peraturan Gubernur nomor 34 Tahun 2023 tentang Petunjuk Teknis Bantuan Keuangan Sarpras TA 2024 https://bit.ly/4bwFgZb 0:45

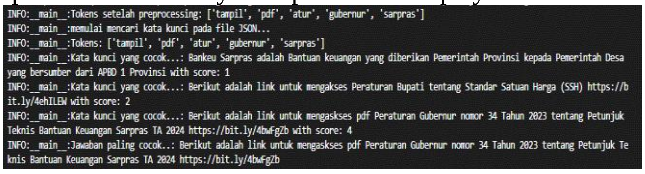
tampilkan sop tentang bankeu sarpras 0:46

Mohon tunggu sebentar, pertanyaan Anda sedang diproses... 0:46

#DINSOSPMD
BerAKHLAK
TIDAK DIPUNGUT BIAYA
STANDAR PELAYANAN
SURAT REKOMENDASI PENYALURAN BANTUAN KEUANGAN SARANA PRASARANA BERSUMBER PADA ANGGARAN PENDAPATAN BELANJA DAERAH PROVINSI (APBD I) KEPADA PEMERINTAH DESA
PERSYARATAN
PRODUK LAYANAN
SURAT REKOMENDASI PENCAIRAN ANGGARAN BANTUAN KEUANGAN
JANGKA WAKTU PELAYANAN
60 MENIT
SISTEM, MEKANISME DAN PROSEDUR
SARANA PENGADUAN
Write a message...

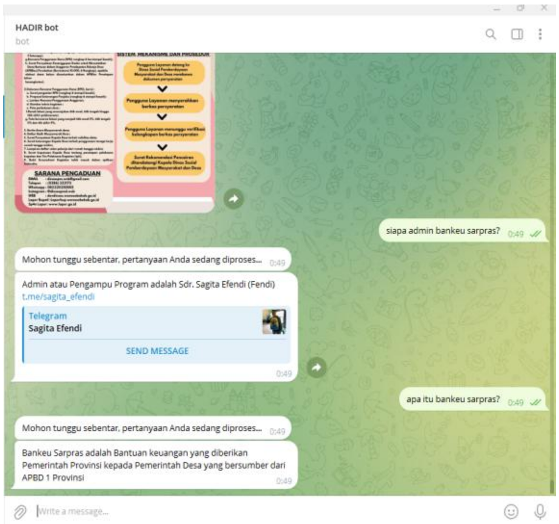
Figure 20 Example of Chatbot Response with Keywords “pdf” and “sop”

Logging the answer search process with the keyword pdf can be displayed in the following image:



```
INFO:__main__:Tokens setelah preprocessing: ['tampil', 'pdf', 'atur', 'gubernur', 'sarpras']
INFO:__main__:memulai mencari kata kunci pada file JSON...
INFO:__main__:Tokens: ['tampil', 'pdf', 'atur', 'gubernur', 'sarpras']
INFO:__main__:Kata kunci yang cocok...: Bankeu Sarpras adalah Bantuan keuangan yang diberikan Pemerintah Provinsi kepada Pemerintah Desa
yang bersumber dari APBD 1 Provinsi with score: 1
INFO:__main__:Kata kunci yang cocok...: Berikut adalah link untuk mengakses Peraturan Bupati tentang Standar Satuan Harga (SSH) https://b
it.ly/4ehILEW with score: 2
INFO:__main__:Kata kunci yang cocok...: Berikut adalah link untuk mengaskses pdf Peraturan Gubernur nomor 34 Tahun 2023 tentang Petunjuk
Teknis Bantuan Keuangan Sarpras TA 2024 https://bit.ly/4bwFgZb with score: 4
INFO:__main__:Jawaban paling cocok..: Berikut adalah link untuk mengaskses pdf Peraturan Gubernur nomor 34 Tahun 2023 tentang Petunjuk Te
knis Bantuan Keuangan Sarpras TA 2024 https://bit.ly/4bwFgZb
```

Figure 21 PDF Keyword Logging



HADIR bot
bot

siapa admin bankeu sarpras? 0:49

Mohon tunggu sebentar, pertanyaan Anda sedang diproses... 0:49

Admin atau Pengampu Program adalah Sdr. Sagita Efendi (Fendi)
t.me/sagita_efendi
Telegram
Sagita Efendi
SEND MESSAGE
0:49

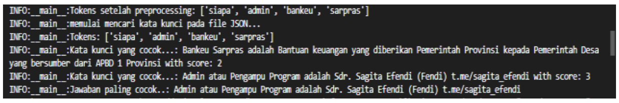
apa itu bankeu sarpras? 0:49

Mohon tunggu sebentar, pertanyaan Anda sedang diproses... 0:49

Bankeu Sarpras adalah Bantuan keuangan yang diberikan Pemerintah Provinsi kepada Pemerintah Desa yang bersumber dari APBD 1 Provinsi 0:49

Figure 22 Example of Chatbot Response for Keywords “who” and “what”

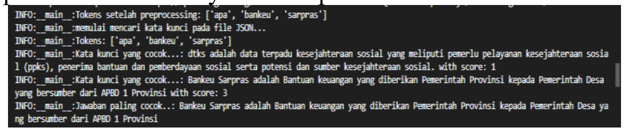
Logging the answer search process with the keyword “who” can be shown in the following image:



```
INFO:__main__:Tokens setelah preprocessing: ['siapa', 'admin', 'bankeu', 'sarpras']
INFO:__main__:memulai mencari kata kunci pada file JSON...
INFO:__main__:Tokens: ['siapa', 'admin', 'bankeu', 'sarpras']
INFO:__main__:Kata kunci yang cocok...: Bankeu Sarpras adalah Bantuan keuangan yang diberikan Pemerintah Provinsi kepada Pemerintah Desa
yang bersumber dari APBD 1 Provinsi with score: 2
INFO:__main__:Kata kunci yang cocok...: Admin atau Pengampu Program adalah Sdr. Sagita Efendi (Fendi) t.me/sagita_efendi with score: 3
INFO:__main__:Jawaban paling cocok..: Admin atau Pengampu Program adalah Sdr. Sagita Efendi (Fendi) t.me/sagita_efendi
```

Figure 23 Logging the Answer Search Process with the Keyword “Who”

Logging the answer search process with the keyword “sop” can be shown in the following figure:



```
INFO:__main__:Tokens setelah preprocessing: ['apa', 'bankeu', 'sarpras']
INFO:__main__:memulai mencari kata kunci pada file JSON...
INFO:__main__:Tokens: ['apa', 'bankeu', 'sarpras']
INFO:__main__:Kata kunci yang cocok...: dtks adalah data terpadu kesejahteraan sosial yang meliputi pemerlu pelayanan kesejahteraan sosia
l (ppks), penerima bantuan dan pemberdayaan sosial serta potensi dan sumber kesejahteraan sosial. with score: 1
INFO:__main__:Kata kunci yang cocok...: Bankeu Sarpras adalah Bantuan keuangan yang diberikan Pemerintah Provinsi kepada Pemerintah Desa
yang bersumber dari APBD 1 Provinsi with score: 3
INFO:__main__:Jawaban paling cocok..: Bankeu Sarpras adalah Bantuan keuangan yang diberikan Pemerintah Provinsi kepada Pemerintah Desa ya
ng bersumber dari APBD 1 Provinsi
```

Figure 24 Logging the Answer Search Process with the Keyword "What"

From the results of several user questions to the chatbot, several answers were produced that were relatively fast, precise and efficient. With almost the same keywords, namely keywords related to "bankeu sarpras" with the subheading method in the JSON file and the use of the TF-IDF method and cosine similarity in the steps of displaying answers to users, the answers displayed to users can be different and very relevant. This means that the HADIR chatbot bot as a virtual assistant in public services and assistance is able to provide accurate answers to users depending on the keywords given to the chatbot, and also depending on the processing of JSON library data.

The following is a table of the accuracy of chatbot answers to users that have been summarized in a table:

Table 1 Chatbot Testing Accuracy Results

No.	Pertanyaan Pada Chatbot	Respon Chatbot	Hasil
1	Pengguna memencet start pada chatbot atau memasukkan perintah “/start”	chatbot memperkenalkan diri dan menjelaskan format inputan	sesuai
2	admin memasukkan perintah “/mode_admin”	bertanya password admin	sesuai
3	admin salah memasukkan password	perintah untuk memasukkan password yang benar	sesuai
4	admin benar memasukkan password	notifikasi password benar, perintah untuk masukkan pertanyaan baru	sesuai
5	admin memasukkan pertanyaan baru	masukkan jawaban baru	sesuai
6	admin memasukkan pertanyaan yang sudah ada pada data library	notifikasi pertanyaan sudah ada, dan respon kepada admin untuk memasukkan pertanyaan baru yang	sesuai
7	admin memasukkan perintah “/cancel”	mode admin berhenti, kembali seperti mode pengguna	sesuai
8	pengguna bertanya dengan kata kunci “sop”	menampilkan gambar sop sesuai kata kunci	sesuai
9	pengguna bertanya dengan kata kunci “pdf”	menampilkan link pdf	sesuai
10	pengguna bertanya dengan kata kunci “siapa kepala bidang”	menampilkan hanya 1 jawaban (jawaban teratas dari file JSON) seharusnya 3 jawaban	kurang sesuai
11	pengguna bertanya dengan pertanyaan “apa itu”	menampilkan 1 jawaban teratas pada file JSON	kurang sesuai
12	pengguna bertanya dengan kata katanya dan kata kunci	menampilkan jawaban sesuai kata tanya dan kata kunci	sesuai

From the results of Table 1, the following accuracy calculations can be obtained.:

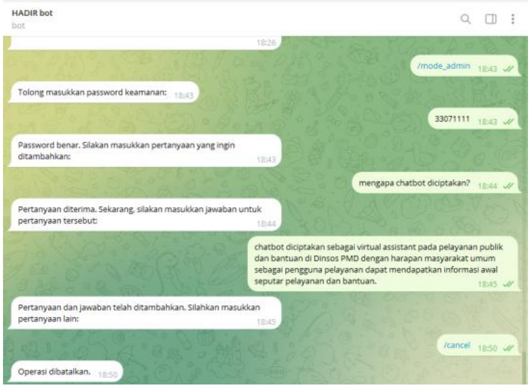
$$\text{accuracy} = \frac{\text{Number of answers appropriate}}{\text{Total number of results}} ,$$

Number of correct answers = 10
Total number of accuracy results = 12

So the chatbot accuracy result =

$$\frac{10}{12} \text{ x } 100\% = 83{,}33\%$$

System Development Method Through Admin Mode. Admin is a user with special access rights who can add new questions to the JSON file via chatbot. This mode cannot be accessed carelessly by ordinary users, the method is to use certain commands on the chatbot. For the HADIR chatbot, the author uses the command "/mode_admin" to be able to get special access rights, to always remember that the writing of the command must be prefixed with a slash "/" so that it is read as a command by the chatbot and not a question sentence. The following is an example of a chatbot response after entering the command "/mode_admin":


HADIR bot
bot

/mode_admin 18:43

Tolong masukkan password keamanan: 18:43

33071111 18:43

Password benar. Silakan masukkan pertanyaan yang ingin ditambahkan: 18:43

mengapa chatbot diciptakan? 18:44

Pertanyaan diterima. Sekarang, silakan masukkan jawaban untuk pertanyaan tersebut: 18:44

chatbot diciptakan sebagai virtual assistant pada pelayanan publik dan bantuan di Dinsos PMD dengan harapan masyarakat umum sebagai pengguna pelayanan dapat mendapatkan informasi awal seputar pelayanan dan bantuan. 18:45

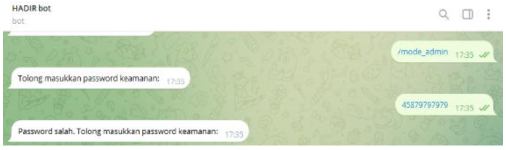
Pertanyaan dan jawaban telah ditambahkan. Silahkan masukkan pertanyaan lain: 18:45

/cancel 18:50

Operasi dibatalkan. 18:50

Figure 25 Chatbot Response When Entering the Command “/mode_admin”

After entering the command “/mode_admin” the chatbot response as seen in figure 27 will submit a response in the form of asking for a security password for the admin, if the admin can correctly enter the right number, then the chatbot response will be followed by “the password is correct, please enter the question you want to add:”, if the password is wrong then the response will appear “wrong password, access denied.” And the user must enter the command again to enter admin mode.


HADIR bot
bot

/mode_admin 17:35

Tolong masukkan password keamanan: 17:35

45879797979 17:35

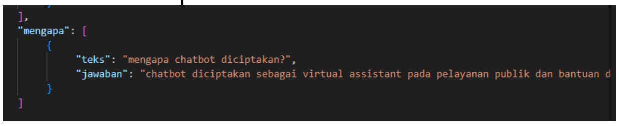
Password salah. Tolong masukkan password keamanan: 17:35

Figure 26 Chatbot Response to Wrong Password

After successful password verification, the admin must enter a new question that does not yet exist in the library data and a response from the chatbot will appear in the form of "the password is correct, please enter a new question to be added." If the question does not yet exist in the library data, the chatbot's response is "the question is accepted, please enter the answer to the question.", if the question entered already exists in the library data, the chatbot's response is "sorry, the question template already exists. Please enter a new question." As in Figure 26.

To exit admin mode, what must be entered is the command "/cancel" and later it will return to being a normal user and can ask the chatbot again as in Figure 27.

The following is data in the form of new questions and answers added to the JSON file:



```
],
"mengapa": [
    {
        "teks": "mengapa chatbot diciptakan?",
        "jawaban": "chatbot diciptakan sebagai virtual assistant pada pelayanan publik dan bantuan d
    }
]
```

Figure 27 New Question and Answer Template in Data Library

4. CONCLUSION

This study successfully built a telegram chatbot as a virtual assistant in public services and assistance at the Dinsos PMD Wonosobo Regency with the TF-IDF method that can operate 24 hours non-stop if run on a server that can help users with existing service information.

With the TF-IDF method and cosine similarity and subheading in the data library, the relevance of the chatbot's answers to user questions can be increased. Because each question processed by the chatbot will be measured for its weight value based on the ranking of each document processed, and the document displayed by the chatbot is the document with the most weight. Based on the results of the chatbot answer accuracy test, 10 correct answers were obtained from a total of 12 types of general questions, so that the results achieved in accuracy were 83.33%.

Adding question and answer templates to the data library must be done frequently with various possible questions or sentences. Adding question and answer templates must also use all lowercase letters to make it easier to search for answer results from the preprocessing method. The more knowledge or knowledge in the data library, the smarter the chatbot will be in answering the questions asked.

5. REFERENCES

A. R. Lahitani, A. E Permanasari, dan A. Setiawan (2016), Cosine Similarity to Determine Similarity Measure: Study case in online essay assessment, Proc. 2016 4th Int. Conf. Cyber. IT Serv. Manag. CITS 2016.

M. Kowsher, M. A. Alam, M.J. Uddin, M. R. Islam, N. Pias, dan A. R. M. Saifullah (2019), Bengali *Informative Chatbot*. *5th Int. Conf. Comput. Commun. Chem. Mater. Electron.* Eng. IC4ME2 2019, pp. 1-7.

M. Mustaqim, A. Gunawan, Y. B. Pratama, I. Zaliman (2023) Pengembangan Chatbot Layanan Publik Menggunakan Machine Learning Dan Natural Language Processing. Jurnal of Information Technologi and society (JITS), vol. 1 (6) pp. 1.

S. Mulyatun., H. Utama., A. Mustopa (2021), Pendekatan Natural Language Processing Pada Aplikasi Chatbot Sebagai Alat Bantu Customer Service. Journal Of Information System Management, Vol. 2 (2), pp. 13.

Sukma, D. Alfian (2019), Klasifikasi Dokumen Temu Kembali Informasi Dengan K-Nearest Neighbour. Jurnal of Chemical and Modelling, 1 (9), pp. 1689-1699