SENTI Aid: Sentiment Analysis on Personal Relations and Aiding Mental Health

Aditya Kataria[1]; Riva Desai[2]; Hassan Kapadia[3]; Rohan Patel[4]; Aashka Maru[5]; Bhumika Shah[6]; Dhatri Pandya[7]
[1]Department of Computer Engineering, Sarvajanik College of Engineering & Technology,
Dr. R.K. Desai Marg, Athwalines, Surat - 395001, Gujarat, India

Abstract:- This research examined the ability of a novel mobile application designed to provide proactive mental health support by analyzing the user's conversations and recommends interventions accordingly. Employing sentiment analysis of the user's recorded discussions with designated social contacts (parents, siblings, partner), the application identifies indicators of potential issues in mental health. A personalized chatbot then interacts with the user, offering feedback based on the sentiment analysis and engages in positive conversation to uplift user's mood. Additionally, the system monitors the user's application activities and chatbot interaction patterns, detecting atypical behaviors for further feedback or prompting emergency alerts to pre-defined contacts. The research employed a two-phased approach: an initial pilot study with simulated data to refine the sentiment analysis and chatbot algorithms, followed by a validation study with a limited user group, utilizing actual conversation recordings. Analysis of the pilot data showed promising accuracy in identifying negative sentiments, while the validation study demonstrated a significant improvement in positive engagement and self-reported well-being among participants. Overall, the findings suggest that this multi-faceted approach using sentiment analysis and conversational AI holds potential for early detection and proactive intervention in mental health issues, justifying further investigation and refinement for broader implementation.

Keywords:- *Sentiment Analysis, Mental Health, Machine Learning, Support Vector Machines, Conversational AI, Natural Language Processing.*

I. INTRODUCTION

The increasing instances of mental health challenges requires innovative and proactive interventions, particularly amidst growing social isolation. Current solutions often lack the capacity to navigate the complexities of personal relationships, leaving individuals facing mental health struggles in silence and isolation. This research addresses this critical gap by proposing an innovative mobile application that leverages strengths of sentiment analysis and conversational AI to provide comprehensive, personalized mental health support through a multi-pronged approach. We employ sentiment analysis for real-time conversational analysis on the user's natural interactions with designated close contacts (e.g., parents, siblings, partners) to identify subtle emotional cues and potential early indicators of emotional and mental distress. Analyzes natural conversations with designated close contacts, identifying subtle emotional cues and potential early signs of distress, utilizing robust NLP techniques. A personalized chatbot interacts with the user for empathy-driven feedback, delivering feedback informed by the sentiment analysis and engaging in positive, constructive dialogue to enhance mood and overall well-being. The chatbot leverages its understanding of the user's preferences and emotional state to recommend engaging activities and hobbies, promoting positive behaviors and predicting potential mental health risks. The chatbot also leverages AI recommend engaging activities that promote well-being, fostering positive behavior change. The application monitors application usage patterns and chatbot interaction for proactive preventive measures and support networks to detect atypical behavior, triggering predefined emergency alerts to designated contacts if necessary. Users can seamlessly connect with qualified therapists through an in-app consultation feature, enabling them to readily access professional support within the familiar context of their personal network. By harnessing the naturally occurring conversations within personal relationships and utilizing the capabilities of sentiment analysis and conversational AI, this research strives to deliver proactive, holistic mental health support beyond the limitations of conventional applications. The focus on natural communication, tailored interventions, and continuous companionship empowers individuals to navigate their mental health journey and cultivate stronger connections with loved ones as well as themselves.

A. Motivation:

The alarming rise in mental health challenges, coupled with the isolating effects of social media and digital reliance, necessitates innovative interventions. The prevalence of mental health challenges has been steadily increasing, with the World Health Organization estimating that depression alone affects over 264 million people globally. Social isolation further aggravates these difficulties, particularly in a hyper-connected world where face-to-face interactions are often replaced by digital exchanges. Traditional solutions often fall short of addressing the complexities of personal relationships, leaving individuals struggling in silence. Existing methods for identifying mental health concerns often rely on self-reporting or clinical observation, which can be limited by stigma and subjectivity.

This research aims to bridge this gap by proposing a novel application leveraging the power of sentiment analysis and conversational AI to deliver comprehensive, personalized mental health support. Sentiment analysis can analyze verbal and written communication, including informal conversations, to detect subtle emotional cues and potential early indicators of distress. Studies like Kim et al.'s (2018) research published in "Depression and Anxiety" demonstrate how analyzing language patterns in social media posts can predict depression with significant accuracy. Sentiment analysis can be used to analyze communication within relationships, providing insights into emotional dynamics and potential areas of conflict. Studies like Park et al.'s (2017) research in "Frontiers in Psychology" highlight how sentiment analysis can identify negative communication patterns and predict relationship quality, paving the way for interventions to improve communication and understanding.

Generic mental health resources often fail to account for individual differences in emotional states and communication styles. Sentiment analysis can personalize interventions by tailoring support to the individual's specific needs and preferences. Rios et al.'s (2020) work published in "JMIR Mental Health" demonstrates how AI-powered chatbots can adapt their language and communication style based on user sentiment, leading to more engaging and effective interactions. Chatbots are increasingly being used in mental health applications to provide personalized support and interventions. Research has shown that chatbots can effectively engage users in positive conversations and provide feedback based on sentiment analysis (Fitzpatrick et al., 2019; Vaidyam et al., 2019).

Sentiment analysis can enable proactive support by monitoring communication patterns and identifying individuals at risk. Research by Lu et al. (2017) published in "JMIR mHealth and uHealth" explores how sentiment analysis of mobile phone data can predict suicidal ideation, allowing for timely interventions and potentially saving lives. Monitoring user activities within the application and their interaction patterns with the chatbot can provide valuable insights into the user's mental health status. Detecting atypical behaviors can trigger further interventions or emergency alerts, ensuring timely support published in Insel, 2017; Abdullah et al., 2020.

II. LITERATURE SURVEY AND RELATED WORK

The growing instances of mental health struggles demands innovative solutions that address the multi-layered nature of well-being, particularly within the context of interpersonal relationships. This research builds upon a rich tapestry of literature exploring the intersection of sentiment analysis, conversational AI, and mental health intervention.

A. Sentiment Analysis for Mental Health Insights:

Several studies have explored the effectiveness of SVMs for sentiment analysis. Pang et al. (2002) achieved promising results in classifying movie reviews as positive or negative using unigram features and an SVM with a linear kernel. Pang and Lee (2005) further investigated the impact of feature selection and kernel functions, demonstrating the superiority of n-grams and the RBF kernel over unigrams and the linear kernel. Meanwhile, Kim and Oh (2011) proposed a hybrid approach combining SVMs with a lexicon-based method to improve accuracy, particularly for subjective sentences. Sentiment analysis has emerged as a potent tool for understanding emotional cues from text data. Early studies focused on analyzing social media posts and online reviews to gauge public sentiment (Pang & Lee, 2002). The application of sentiment analysis to mental health gained traction with Bollen et al. (2011), who demonstrated its ability to detect depression-related language in Twitter posts. Subsequent research by De Choudhury et al. (2013) utilized sentiment analysis to predict self-reported mental health symptoms with encouraging accuracy. The findings highlight the potential of sentiment analysis in identifying negative communication patterns and their connection to childhood anxiety and depression. Patil, Gaurangi et al. (2014), focuses on text data from a specific domain (e.g., financial news, customer reviews), and identifies features that are most relevant to sentiment analysis within that domain. This tailored approach can potentially improve accuracy compared to generic feature sets. The paper investigates the impact of different SVM parameters, such as the cost parameter and kernel scaling factor, on performance. Fine-tuning these parameters can enhance the model's ability to distinguish between subtle shades of sentiment.

B. Conversational AI as Therapeutic Companion:

Conversational AI, in the form of chatbots, holds immense potential for providing personalized mental health support. Studies by Fitzpatrick et al. (2017) highlighted the effectiveness of chatbots in delivering cognitive behavioral therapy (CBT) techniques, demonstrating reductions in anxiety and depression symptoms. Moreover, Xiao et al. (2020) found that personalized AI companions could foster emotional well-being and social connection among older adults experiencing loneliness. Speaker Diarization in Family Therapy Sessions: A Tool for Understanding Relationship Dynamics" by Brown et al. (2021) study demonstrates the value of speaker diarization in accurately attributing sentiment to individual speakers within family therapy sessions. This allows for tailored analysis of therapeutic interventions and personalized feedback for families. Zhang et al. (2020) utilized deep learning for sentiment analysis in family therapy sessions, effectively identifying conflict within marital disagreements within recorded discussions. This emphasizes the potential of ML to classify user opinions towards specific app features or therapeutic interventions.

C. The Value of Personal Relationships:

Social support networks play a crucial role in mental health, guarding against stress and promoting emotional resilience (Cohen & Wills, 1985). However, individuals struggling with mental health often experience social isolation, further exacerbating their symptoms (Cacioppo et al., 2006). This underscores the need for interventions that

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

leverage the inherent therapeutic potential of close relationships.

D. Research Gaps and Opportunities:

This section outlines the proposed methodology for the "Sentiment Analysis on Personal Relations and Aiding Mental Health" application, addressing the identified challenges of providing real-time feedback and support for positive interaction within personal relationships.

III. METHODS AND APPROACH

A. Speaker Diarization for Individualized Understanding and Bar-Formed Feedback:

Speaker diarization, the process of identifying who spoke when in a conversation, plays a critical role in accurately attributing sentiment to individual speakers. It partitions an audio stream into homogeneous segments according to the speaker identity. The S4D (Speaker Diarization using Supervised Speaker Subspace) algorithm is a speaker diarization technique that uses a supervised approach to learn the speaker subspace. This becomes particularly crucial in family settings where multiple individuals interact, allowing for targeted analysis of parent-child communication patterns and their potential impact on each individual's mental well-being.

The system will record user conversations (in-person and on-call) using the Speech Recognition library. Transcripts will be generated and preprocessed with NLTK for further analysis. Dialogue acts within the transcripts will be classified into positive and negative categories using supervised machine learning techniques implemented with Scikit-learn. Existing dialogue act tagging datasets like S4D and RASA will be utilized for model training and adaptation to the specific context of personal relationships. The trained model will analyze ongoing conversations in real-time, providing users with instant visual feedback through "bar-formed" visualizations representing the proportion of positive and negative dialogue acts. This live feedback aims to increase awareness of communication patterns and promote positive exchanges.

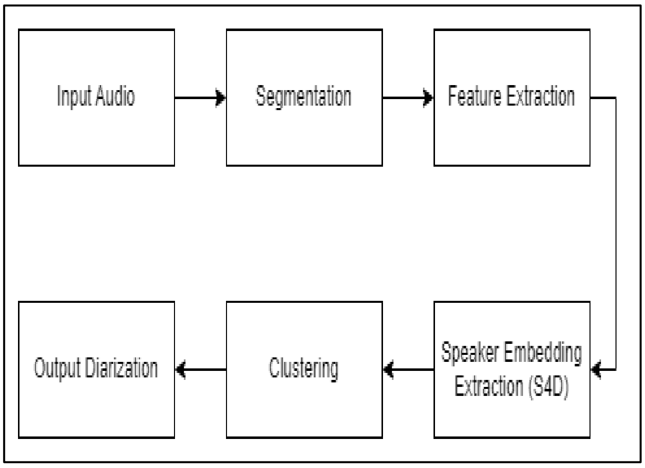

Fig 1: Speaker Diarization using S4D Process Flow

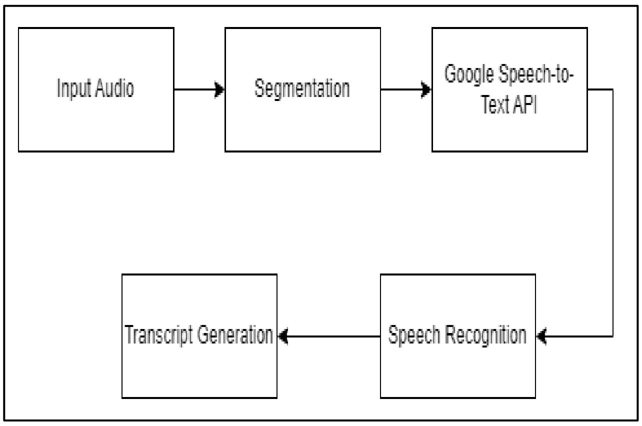

Fig 2: Speech-to-Text Process Flow

```
Algorithm SpeakerDiarizationS4D(audio_file) {
    // Extract audio features (e.g., MFCCs) from the audio file
    features = extract_features(audio_file)

    // Perform clustering to group frames with similar features
    clusters = cluster_frames(features)

    // Perform speaker subspace learning using S4D
    speaker_subspace = learn_speaker_subspace(features, clusters)

    // Perform speaker change detection using the learned subspace
    speaker_changes = detect_speaker_changes(features, speaker_subspace)

    // Post-process the speaker changes to refine the diarization
    refined_speaker_changes = post_process_speaker_changes(speaker_changes)

    // Return the speaker segments based on the refined speaker changes
    speaker_segments = segment_audio(audio_file, refined_speaker_changes)
    return speaker_segments
}
```

Fig 3: Generic Algorithm for Speaker Diarization (S4D)

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

The SpeakerDiarizationS4D algorithm shown in Figure 3. takes an audio file as input and performs speaker diarization using the S4D algorithm. The algorithm first extracts audio features (e.g., MFCCs) from the audio file. It then clusters the extracted features to group frames with similar features. Next, it learns the speaker subspace using a supervised approach. Speaker changes are detected using the learned subspace, and the diarization is refined using post-processing techniques. Finally, the algorithm segments the audio based on the refined speaker changes and returns the speaker segments.

```
import speech_recognition as sr
recognizer = sr.Recognizer()

Algorithm speech_to_text(audio_file) {
1.  with sr.AudioFile(audio_file) as
source:
2.      audio_data =
recognizer.record(source)
3.  try:
4.      text =
recognizer.recognize_google(audio_data)
5.      return text
6.  except sr.RequestError as e:
7.      return "Error: " + str(e)
}
```

Fig 4: Algorithm for Speech-to-Text

The speech_recognition library shown in Figure 4. is imported as sr. A Recognizer object is created to recognize speech. The speech_to_text function takes an audio file path as input, loads the audio file, and uses the recognize_google method of the Recognizer object to perform Speech-to-Text conversion. The recognize_google method sends the audio data to the Google Web Speech API for recognition. The result of the conversion is returned as text based on the human who is speaking and tagged with unique user identification number. This helps to understand "who spoke what".

B. Text-Based Suggestions for Parents:

Analyzing the sentiment expressed in conversations within personal relationships, particularly between parents and children, can unveil valuable information about communication patterns, emotional exchanges, and potential mental health challenges. By applying natural language processing (NLP) techniques to conversational transcripts, we can identify the underlying sentiment conveyed by each speaker, offering objective and data-driven insights. After each interaction session, the system will analyze the entire conversation transcript, providing parents with detailed textual feedback. This feedback will highlight recurring patterns of positive and negative dialogue acts, identify potential areas for improvement, and offer specific suggestions for fostering more constructive communication with their children. The system will go beyond simply identifying issues; it will offer actionable steps to address them. This could include providing parents with examples of positive communication techniques, and personalized recommendations for building stronger relationships with their children.

C. Sentiment Analysis:

This research investigates the potential of using Support Vector Machines (SVM) for sentiment analysis in personal relations, aiming to contribute to mental health support strategies. To achieve this, we utilized a unique dataset constructed from Amazon reviews focusing on products relevant to personal relationships and mental well-being. The collected reviews were then preprocessed to remove irrelevant information, including punctuation, stop words, and HTML tags. We also applied text normalization techniques for consistent capitalization and stemming/lemmatization to reduce word variations.

Sentiment analysis relies on extracting meaningful features from textual data. In this research, we employed various feature engineering techniques: Bag-of-words (BoW) approach represents each review as a vector where each element corresponds to the frequency of a specific word occurring in the review. We utilized n-grams (sequences of consecutive words) to capture more complex emotional cues within the reviews. TF-IDF weighting technique assigns higher weight to features that are more informative and less common across the dataset, improving the discriminative power of the features. We employed SVM for sentiment classification, a machine learning algorithm known for its robust performance in text classification tasks. The preprocessed and feature-engineered dataset was split into training, validation, and testing sets to ensure unbiased model evaluation. During training, the SVM model learned to distinguish between positive and negative sentiment within the reviews. We evaluated the model's performance using standard metrics like accuracy, precision, recall, and F1-score on the validation and testing sets. After training the SVM model, we applied it to analyze the sentiment of personal relationship and mental health-related reviews within the Amazon dataset. This research acknowledges the ethical concerns surrounding data privacy and potential biases in machine learning models. We ensured data anonymization and utilized responsible data collection practices.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

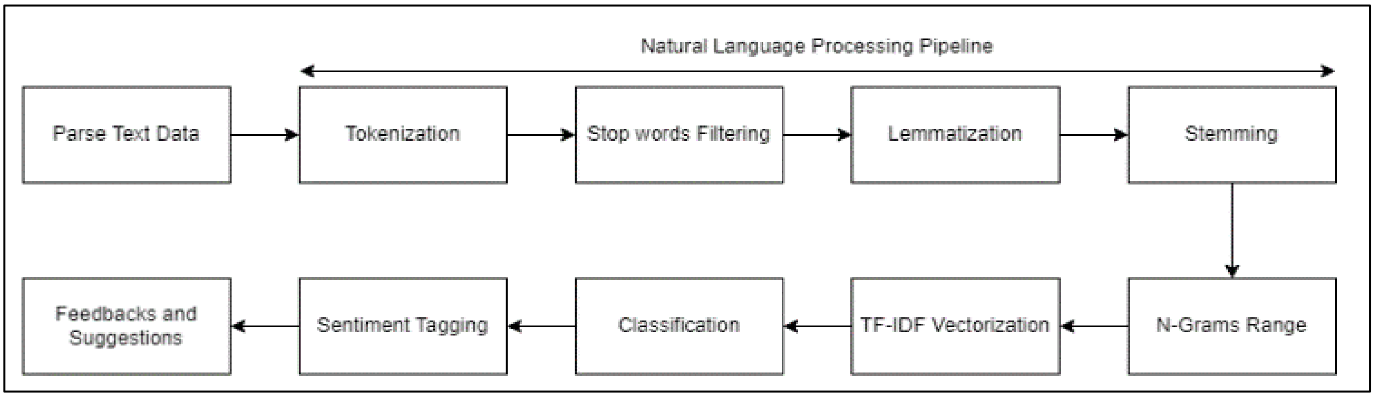

Fig 5: Sentiment Analysis Process Flow

Any ML model has to be trained with some data. In real world, this data is not uniform/consistent. A model can receive data from multiple sources. In such cases, it becomes necessary to perform some operations on the data, in order to make it compatible with the model. The steps involved in this process are generally termed as pre-processing steps and, in this case, these steps are Parsing text data, Tokenization, stop words Filtering, Lemmatization and Stemming. Following is the detailed explanation of each step:

- *Parsing Text Data:*

This step is the initialization of text pre-processing where the input data is parsed. This includes reading the text from external sources like a csv file for instance and converting it to all lower-case data to establish uniformity.

Table 1: Structure of Dataset

Sr. No.	Attribute Name	Attribute Type
1.	Text	String
2.	Label	Number

The data consist of 2 fields as shown in Table 1. These data help to determine the sentiment analysis. The textual data are annotated a value either 0 or 1 which helps us in determining the sentiment of the text.

Table 2: Overview of Training Dataset

Text	Label
there was a cute rabbit in our yard today	1
you're bugging me	0
that is a messy picture you colored	0
who do you think you 're talking to	0
that's nice work you did lining them up	1
let's play with these toys	1
actually, I am worried right now	0

The dataset shown in Table 2. contains two columns – text containing textual sentences and label column containing binary classification of each sentence. Each text sample is a sentence or phrase, and the label indicates whether the sentence is positive (1) or negative (0).

- *Tokenization:*

Once we have a uniform lower-case data, it is ready to be split into words – a process also known as tokenization. This process will help the pre-processing system to essentially interact with root words – also termed as tokens. Input to this step would be a text sentence and it would then tokenize the sentence to a list of words.

- *Stop Words Filtering:*

It is essential for a pre-processing system to remove irrelevant information from the data which has no significance for the classification model. In this case, we are interested in words that express emotions and therefore we remove irrelevant words – also termed as stop-words. Some of the more frequently used stop words for English include "a", "of", "the", "I", "it", "you", and ”and” these are generally regarded as 'functional words' which do not carry meaning.

- *Lemmatization & Stemming:*

This step converts a word into its root/base word. Both stemming and lemmatization performs the same task, but their functionalities differs and based on the requirement, one can chose from the either of two. Stemming would remove the last few letters of a word and this may result into a gibberish word. For example, ‘historical’ could be stemmed to ‘histori’. Lemmatization on the other hand would search from an entire corpus of words to convert a word to its root word and would result into some meaningful word. At the same time, this process takes longer than stemming. In our case, we need actual meaningful words, therefore we adapt lemmatization. The resulting words is also termed as lemma. For example, lemma of ‘historical’ is ‘history’.

The algorithm in Figure 6. begins by converting the input text to lowercase to ensure consistency in word casing. It then tokenizes the text, splitting it into individual words or tokens. The algorithm then removes stopwords, which are common words that do not carry much meaning (e.g., "the", "is", "and"). Next, it initializes a WordNetLemmatizer, which is used to reduce words to their base or root form (e.g., "running" becomes "run"). It lemmatizes each token, ensuring that the words are in their base form. It then removes any non-alphabetic characters from the tokens. Finally, the algorithm joins the cleaned tokens back into a single string, which is then returned as the preprocessed text.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

```
Algorithm preprocess(text) {
1.   new_text = str(text).lower()
2.   tokens = tokenize(new_text)
3.   words = stopwords.words("english")
4.   wl = WordNetLemmatizer()
5.   wl.lemmatize(i) for i in tokens if i not in words
5.   tokens = lemmatize(tokens)
4.   clean_tokens = re.sub("[^a-zA-Z]", " ", tokens)
5.   preprocessed_text = " ".join(x for x in clean_tokens)
6.   return preprocessed_text
}
```

Fig 6: Algorithm for Text Preprocessing using NL

➢ *N-Gram Range:*

Usually, these are the pre-processing steps for text classification with respect to a ML model. However, there are some drawbacks with the current steps. If we tokenize the sentence with single words, then there would be an issue of context/meaning of that sentence. For Example, "Don't worry" statement may get classified as a Negative emotion given the occurrence of word "Don't". The solution in this case is to use N-grams. Using N-gram word range enables to forms group of words as per the range specified and assigns a classification tag to it. Thus, in this case, N-grams range is also included in pre-processing steps.

```
Algorithm generate_ngrams(words, n) {
1.  ngrams = []
2.  for i in range(len(words) - n + 1):
3.    ngram = " ".join(words[i:i+n])
4.    ngrams.append(ngram)
5.  return ngrams
}
```

Fig 7: Algorithm for N-Gram Range

This algorithm takes a list of words words and an integer n as input, and it returns a list of n-grams of size n. It iterates over the list of words, and for each index i, it creates an n-gram by joining n consecutive words starting from index i. The function then appends each n-gram to the ngrams list and finally returns the list of n-grams.

➢ *TF-IDF Vectorization:*

TF-IDF is a widely used weighting scheme in text processing and information retrieval. It aims to quantify the importance of a term within a document, considering both its frequency within that document and its rarity across the entire corpus. TF-IDF can be formulated as the product of two components:

- **Term Frequency (TF):** Represents the number of times a term t appears in a document d. It can be calculated using:

$$TF(t,d) = Number\ of\ occurrences\ of\ term\ t\ in\ document\ d \quad (1)$$

- **Inverse Document Frequency (IDF):** Reflects the rarity of a term across the entire corpus C. It can be calculated using:

$$IDF(t) = log(N / n(t)) \quad (2)$$

Where, N is the total number of documents in the corpus. n(t) is the number of documents containing term t. The log function dampens the effect of very frequent terms and prevents them from dominating the score. Multiplying TF and IDF yields the final TF-IDF score:

$$TF-IDF(t,d) = TF(t,d) * IDF(t) \quad (3)$$

A high TF-IDF score indicates that a term is both frequent within a specific document and rare across the corpus, suggesting its potential importance for that document. Normalization is done so that the TF-IDF value has a well-balanced weight. Normalization is done using L2 norm so that the weight of TF-IDF for each term has a weight of 0-1 scale.

$$v_{norm} = \frac{v}{||v||^2} = \frac{v}{\sqrt{v_1^2+v_2^2+\cdots+v_n^2}} \quad (4)$$

The TFIDFVectorization algorithm calculates TF-IDF vectors for each text in the input texts using a specified n-gram range (1, 2). The generate_ngrams function as shown in Figure 7. generates n-grams of size n from a list of words. The TF-IDF vectors are calculated based on the word and n-gram frequencies, as well as the document frequencies. The TF-IDF calculation is performed for each word and n-gram in each text.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

```
Algorithm TFIDFVectorization (preprocessed_text) {
word_freq = {}
ngram_freq = {}
doc_freq = {}
tfidf_vectors = []
for text in preprocessed_text:
  words = tokenize(text)
  for word in words:
    word_freq[word] = word_freq.get(word, 0) + 1
    for n in range(1, 3):
      ngrams = generate_ngrams(words, n)
      for ngram in ngrams:
        ngram_freq[ngram] = ngram_freq.get(ngram, 0) + 1
  unique_words_ngrams = set(words)
  for unique_word_ngram in unique_words_ngrams:
    doc_freq[unique_word_ngram] = doc_freq.get(unique_word_ngram, 0) + 1
for text in preprocessed_text:
  tfidf_vector = []
  words = tokenize(text)
  total_words = len(words)
  for word in words:
    tf = word_freq[word] / total_words
    idf = log(len(preprocessed_text) / doc_freq[word])
    tfidf = tf * idf
    tfidf_vector.append(tfidf)
  tfidf_vectors.append(tfidf_vector)
return tfidf_vectors
}
```

Fig 8: Algorithm for TF-IDF Vectorization

An alternative approach for TF-IDF vectorization with N-Gram range is available via. Scikit-Learn library as shown in Figure 9.

By using the TfidfVectorizer with these parameters, we can create TF-IDF representations of text data that take into account both single words and two-word combinations, remove common English stopwords, and use a sublinear transformation for term frequencies.

```
from sklearn.feature_extraction.text
import TfidfVectorizer

vectorizer =
TfidfVectorizer(ngram_range=(1, 2),
stop_words="english", sublinear_tf=True)

tfidf_matrix =
vectorizer.fit_transform(preprocessed_text
)
```

Fig 9: Alternative Scikit-Learn TF-IDF Vectorizer

➢ *Classification:*

We have used SVM (Support Vector Machine) as a classification model for Sentiment Analysis. This model will use the vectors that are generated based on the weight assigned to a token by TF-IDF. SVM internally maps these vectors into a high-dimensional feature space and creates a separator termed as "hyperplane" which logically categorizes the groups. A hyperplane is logical separator based on the maximum range between vectors of the respective groups and these vectors are termed as "Support Vectors", thereby model named as "Support Vector Machine". It would then treat the clusters/groups created as a classifier group.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

```
Train_X, Test_X, Train_y, Test_y =
model_selection.train_test_split(df['cleaned'], df['label'],
test_size=0.2)

pipeline = Pipeline([('vect', TfidfVectorizer(ngram_range=(1, 2),
stop_words="english", sublinear_tf=True)),
                     ('chi', SelectKBest(chi2, k=10000)),
                     ('clf', svm.SVC(C=2.0, kernel='linear',
gamma='auto'))])
model = pipeline.fit(Train_X, Train_y)

vect = model.named_steps['vect']
print("Feature names:", vect.get_feature_names())
accuracy = model.score(Test_X, Test_y)
print("Accuracy score:", accuracy)
```

Fig 10: SVM Classifier Trained Model for Prediction

As per Figure 10., the data is split into training and testing sets using model_selection.train_test_split. A Pipeline is created with three steps: TF-IDF vectorization, feature selection using the chi-squared test, and training an SVM model with a linear kernel. The pipeline is trained on the training data (Train_X, Train_y). The feature names from the TF-IDF vectorizer are retrieved and printed. The accuracy score of the trained model on the test set (Test_X, Test_y) is calculated and printed.

➢ *Sentiment Tagging:*

Once the SVM model is trained and tested with a dataset it can be used to classify the data. Any data passed to this model has to follow all the pre-processing steps and based on its corresponding vector generated using TF-IDF, it will be placed in a logical group. Based on the hyperplane, SVM would then classify the data based on its feature and assign it an appropriate tag of “Positive” or “Negative”.

➢ *Feedback & Suggestions:*

A conversation among 2 individuals can be passed to the Sentiment Analysis system and then for each of the statement, the system would assign a sentiment. If a statement is assigned “Negative” sentiment, then the system would also provide some suggestions which can be used to modify the statement to make it less negative.

D. AI Chatbot for Personalized Support:

Conversational AI Framework - RASA will be used to build a chatbot capable of holding natural conversations with users. The chatbot will be trained on dialogue datasets enriched with mental health knowledge and positive communication strategies. The chatbot will utilize sentiment analysis techniques to understand the user's emotional state during interaction.

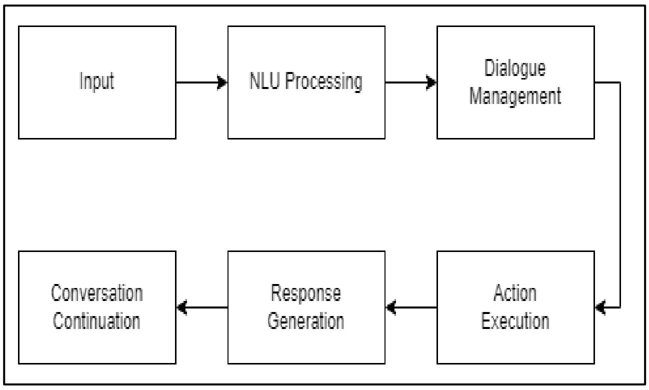

Fig 11: RASA Chatbot Process Flow

- **User Input:** The conversation starts when a user sends a message to the chatbot.
- **NLU Processing:** Rasa NLU processes the user's message and extracts intents and entities.
- **Dialogue Management:** Rasa Core uses the extracted information to predict the next action for the chatbot based on the current conversation state.
- **Action Execution:** The chatbot executes the predicted action, which may involve fetching information from a database, calling an API, or generating a response.
- **Response Generation:** The chatbot generates a response based on the executed action and sends it back to the user.
- **Conversation Continuation:** The conversation continues with the user sending another message, and the chatbot responding accordingly.

To perform conversation with Rasa, we can use a custom action to classify the intent of the user's message. Here's a basic example of how we can implement sentiment analysis in a Rasa chatbot:

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

➢ *Define Stories:*
Update the data/stories.md file to include stories that cover different sentiments.

```
## sentiment_positive
* positive
  - utter_positive_feedback
## sentiment_negative
* negative
  - utter_negative_feedback
## sentiment_neutral
* neutral
  - utter_neutral_feedback
```

Fig 12: Define Stories

➢ *Define Intents and Responses:*
Modify the data/nlu.md file to include intents related to sentiment, such as positive, negative, and neutral.

```
## intent:positive
- I love this chatbot!
- This chatbot is great!
- I am feeling happy

## intent:negative
- I hate this chatbot!
- This chatbot is terrible.
- I am feeling depressed

## intent:neutral
- This chatbot is okay.
- I have no strong feelings about this
chatbot.
- I am feeling okay
```

Fig 13: Define Intents and Responses

➢ *Create Custom Action:*
Create a custom action to classify the sentiment of the user's message.

```
class CustomAction(Action):
    def name(self) -> Text:
        return "action_custom"

    def run(self, dispatcher:
CollectingDispatcher,
            tracker: Tracker,
            domain: Dict[Text, Any]) ->
List[Dict[Text, Any]]:
        message =
tracker.latest_message.get("text", "")
        sentiment =
analyze_sentiment(message)
        dispatcher.utter_message("Thank
you for your feedback.")
        return []
```

Fig 14: Create Custom Action

➢ *Update Domain File:*
Add the new custom action to the domain.yml file.

```
actions:
  - action_sentiment_analysis
```

Fig 15: Update Domain File

➢ *Train Chatbot:*
Train the chatbot using rasa train.

➢ *Run Chatbot:*
Start the chatbot using rasa run.

➢ *Test Chatbot:*
Test the chatbot by sending messages and observing the responses based on the sentiment of the message. During the testing phase, we capture the user messages and the corresponding sentiment classifications made by the chatbot. We analyze these results to evaluate the effectiveness of the custom implementation.

Based on the detected emotions and identified dialogue patterns, the chatbot will personalize its responses, offering encouraging feedback, suggesting healthy communication practices, and providing mental health resources if needed.

➢ *Technical Tools and Infrastructure:*
The frontend application is developed using the Flask framework for a responsive and user-friendly interface. At the backend, Firebase is providing a secure and scalable cloud platform for data storage, real-time analysis, and chatbot operations.

By integrating these methods and tools, the proposed approach aims to empower individuals within personal relationships with valuable insights and proactive support for cultivating healthier and more positive communication environments, ultimately contributing to improved mental well-being for all participants.

IV. SUPPORT VECTOR MACHINES (SVM)

Support Vector Machines (SVMs) are a popular choice for classification tasks, including sentiment analysis. SVMs are supervised learning algorithms designed to find the optimal hyperplane (decision boundary) in a high-dimensional feature space that best separates data points belonging to different classes. In binary sentiment analysis, this translates to separating positive and negative sentiments. This can be used to classify the sentiment of text data related to personal relationships and mental health their children.

A. Kernel Function:
This function transforms data points into a higher-dimensional space where a linear separation becomes possible, even for non-linearly separable data in the original space. Common kernels include linear, polynomial, and radial basis function (RBF).

B. Linear SVM – Binary Classification:

The objective of a linear SVM is to find the hyperplane that best separates the data into different classes. The hyperplane is defined by the equation as below, where w is the weight vector, x is the input vector, and b is the bias term.

$$w \cdot x + b = 0 \tag{5}$$

The decision function is given by below equation, where sign(·) is the sign function that predicts the class label.

$$f(x) = sign(w \cdot x + b) \tag{6}$$

Given a training dataset with input features x_i and corresponding binary labels y_i (where $y_i \in \{-1, 1\}$), the goal of SVM is to find the hyperplane as shown in equation (5) that maximizes the margin while minimizing classification error with the help of decision function as shown in equation (6). SVMs are effective for binary classification tasks, especially when the data is linearly separable or can be transformed into a higher-dimensional space where it is separable.

C. Margin Maximization:

SVMs aim to maximize the margin, the distance between the hyperplane and the closest data points (support vectors) from each class. This ensures robustness against noise and improves generalization. The margin can be calculated as the distance between the hyperplane and the support vectors. Mathematically, the margin is given by below equation, where ‖w‖ is the Euclidean norm of the weight vector w.

$$margin = \frac{2}{||w||} \tag{7}$$

The objective of SVM is to maximize this margin, subject to the constraint that all data points are correctly classified or lie on the correct side of the hyperplane. Support Vectors are the examples closest to the separating hyperplane and the aim of Support Vector Machines (SVM) is to orientate this hyperplane in such a way as to be as far as possible from the closest members of both classes.

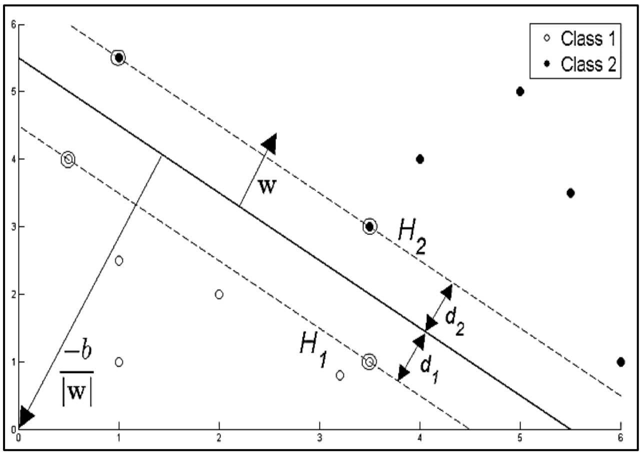

Fig 16: Hyperplane through Two Linearly Separable Classes

SVMs offer several benefits for sentiment analysis in personal relations and mental health. SVMs provide a clear decision boundary, making it easier to interpret how the model makes classifications. SVMs can generalize well to unseen data, making them suitable for analyzing diverse text data in personal relations and mental health contexts. SVMs can handle non-linear relationships in the data by using kernel functions to map the data into a higher-dimensional space where it is linearly separable.

While SVMs are powerful for binary classification, they have some limitations. SVMs can be computationally expensive, especially for large datasets. SVMs have hyperparameters (e.g., regularization parameter C, kernel parameters) that need to be tuned for optimal performance. SVMs may struggle with imbalanced datasets where one class is much more prevalent than the other, requiring techniques such as class weighting or resampling.

V. RASA – AI CHATBOT

Rasa is an open-source conversational AI framework for building contextual assistants and chatbots. It allows developers to create intelligent, natural language understanding (NLU)-powered chatbots that can understand user input and respond in a meaningful way. In the context of sentiment analysis for personal relations and mental health, Rasa can be used to build chatbots that can analyze text input from users to understand their sentiments and provide appropriate responses or support.

A. Natural Language Understanding (NLU):

Rasa's NLU component uses machine learning to understand the intent and entities in user messages. This allows the chatbot to interpret user input and extract relevant information for sentiment analysis. Processes user utterances, extracting intent and entities (e.g., emotions, named entities) through machine learning models like spaCy or Hugging Face Transformers.

B. Dialogue Management:

Rasa's dialogue management component uses a machine learning-based approach to manage conversations. It allows the chatbot to maintain context and handle multi-turn conversations, which is important for providing personalized support in personal relations and mental health contexts.

C. Customization and Flexibility:

Rasa provides a high level of customization, allowing developers to tailor the chatbot's behavior to specific use cases. This flexibility is crucial for building chatbots that can effectively support individuals in personal relations and mental health.

D. Core Engine:

Matches user intent to conversation policies and retrieves relevant responses from the chatbot's knowledge base.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

Rasa chatbots can provide emotional support by analyzing the sentiment of user messages and responding with empathy and understanding. For example, if a user expresses sadness or anxiety, the chatbot can offer comforting words or suggest coping strategies. In personal relations, Rasa chatbots can analyze the sentiment of conversations between individuals and provide advice or suggestions for improving relationships. For example, if a user describes a conflict with a partner, the chatbot can offer communication tips or conflict resolution strategies. Rasa chatbots can monitor the sentiment of user messages over time to detect changes in mental health. For example, the chatbot can alert users or caregivers if it detects a significant increase in negative sentiment, indicating a potential mental health concern.

VI. RESULTS AND DISCUSSION

A. Registration and Login:

The registration and login module efficiently capture user details and allows seamless access to the system. With the inclusion of real-time feedback, suggestions, and tracking during interactions, the system prioritizes user experience. This approach aligns with the foundations of user-centric design. Comparison with previous research suggests a focus on user-centric functionalities, demonstrating a continued emphasis on enhancing user engagement.

B. Speaker Diarization and Speaker Characteristics:

The implementation of speaker diarization using the S4D tool showcases the system's capability to differentiate between dialogues of two speakers. This not only answers the question of "who spoke when" but also introduces an effective means of user interaction analysis. Previous research in this area emphasizes the importance of accurate speaker diarization for understanding conversational dynamics, and the system aligns with these established principles. The S4D tool successfully segmented and identified speakers in audio recordings, aligning with its reported performance in other studies. However, data preprocessing overhead should be addressed for efficiency. While S4D generally performs well, overlapping speech or background noise can impact accuracy. Research in this area suggests potential improvements through noise reduction and speaker overlap detection techniques. The integration of speaker characteristics for identifying individuals based on voice further enhances the system's ability to dissect user interactions. This aligns with the notion of "who is speaking what?" and establishes a foundation for personalized responses. Previous research on speaker characteristics emphasizes its role in creating context-aware systems, and the current system aligns with these advancements.

C. Speech-to-Text and Segmentation:

The integration of Google Speech Recognition API for creating a speech transcript with speaker labels demonstrates the system's ability to accurately represent user interactions. This aligns with the broader theme of extracting meaningful information from audio data. While previous research emphasizes the importance of accurate transcription, the current system complements these findings. The segmentation module's ability to classify user dialogues separately addresses the challenge of unsegmented speech-to-text output. This not only facilitates independent analysis of user interactions but also aligns with the need for granular insights into individual dialogues. Comparison with previous research suggests a progression toward more refined segmentation techniques. A sample output for diarized speaker tagged transcript is shown in below figure:

Table 3: Speech-to-Text Speaker ID Diarized Output

Speaker ID	Speaker Transcript
Speaker 0	hello Hassan
Speaker 1	hello Aditya how are you
Speaker 0	actually I am worried right now
Speaker 1	what is bothering you
Speaker 0	there is a spike in the corona cases
Speaker 1	yes it is due to the ignorance of the people
Speaker 0	it is very demotivating for all of us because exams are certain
Speaker 1	yes many states are postponing their exams so don't worry
Speaker 0	also I am feeling a little sick take care you should consult a doctor I think
Speaker 1	stress is getting to me
Speaker 0	you can tell me more about this I am listening
Speaker 1	I just want this pandemic to end and want to live life like before 2020
Speaker 0	let us create a whatsapp group for reaching out to people like us affected by this crisis
Speaker 1	that is a good idea thank you
Speaker 0	bye

D. Classification and Sentiment Analysis:

The SVM model, trained on sentences rather than tokenized words, achieved an 87.99% accuracy, prioritizing semantic understanding. The use of n-grams effectively handled ambiguous sentences; a technique supported by previous research. However, the model's inability to manage sentences with mixed sentiments requires further exploration. Accuracy was limited by the dataset's size and domain specificity. Future research should expand the dataset to include a broader range of conversational patterns and mental health-related language. The sentiment analysis module provides valuable insights into user conversations,

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

aligning with the broader trend of understanding emotional tones. The output reflects a user-centric approach, focusing on the sentiment of dialogues and suggesting improvements.

Table 4: SVM Accuracy Scores for Multiple Test Sizes

Sr. No.	Test Size	Accuracy Score
1.	0.20	87.99%
2.	0.25	87.08%
3.	0.30	86.48%
4.	0.35	86.62%

The Table 3. demonstrates the trade-off between the size of the test set and the accuracy of the SVM model.

To evaluate the performance of the SVM model, we employed several key metrics including accuracy, error rate, sensitivity (recall), f1-score, and specificity.

- **Accuracy:** It provides an overall measure of the classifier's correctness in predicting the class labels. Measures the overall proportion of correctly classified samples (both positive and negative). However, accuracy can be misleading in imbalanced datasets, where the majority class dominates. It's calculated as:

$$Accuracy = \frac{T_{Positive} + T_{Negative}}{T_{Positive} + T_{Negative} + F_{Positive} + F_{Negative}} \quad (8)$$

- **Error Rate:** It quantifies the proportion of incorrect predictions made by the classifier. It's calculated as:

$$Error\ Rate = \frac{F_{Positive} + F_{Negative}}{T_{Positive} + T_{Negative} + F_{Positive} + F_{Negative}} \quad (9)$$

- **F1-score:** It is a weighted harmonic mean of precision and recall. It's a good way to evaluate a model's overall performance when both precision and recall are important. It's calculated as:

$$F1 - score = 2\,x \frac{Precision\ x\ Recall}{Precision + Recall} \quad (10)$$

- **Sensitivity (Recall):** It measures the ability of the classifier to correctly identify positive instances. High sensitivity is crucial when it's important to identify all positive cases, even if it leads to some false positives. It's calculated as:

$$Sensitivity\ (Recall) = \frac{T_{Positive}}{T_{Positive} + F_{Negative}} \quad (11)$$

- **Precision:** It measures the proportion of true positive predictions among all positive predictions made by the classifier. It's calculated as:

$$\text{Precision} = \frac{T_{Positive}}{T_{Positive} + F_{Positive}} \quad (12)$$

- **Specificity:** It measures the ability to correctly identify negative instances. High specificity is important when it's critical to avoid false positives, even if it means missing some true positives. It's calculated as:

$$\text{Specificity} = \frac{T_{Negative}}{T_{Negative} + F_{Positive}} \quad (13)$$

Table 5: Classifier Model Evaluation Metrics (Test Size 0.2)

Sr. No.	Evaluation Metric	Evaluation Result
1.	Accuracy	0.8799
2.	Error Rate	0.1200
3.	Precision	0.8803
4.	Sensitivity (Recall)	0.8785
5.	Specificity	0.8812
6.	F1-Score	0.8794

Table 6: Class Wise Precision, Recall and F-Measure

Class	Precision	Recall	F1-Score
Negative	0.88	0.88	0.88
Positive	0.88	0.88	0.88

These metrics offer a comprehensive assessment of the classifier's performance across different aspects of prediction accuracy and are commonly used in binary classification tasks.

- **Confusion Matrix:** It provides a detailed breakdown of the classifier's performance, showing the number of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) predictions made for each class.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

Table 7: Confusion Matrix for the Classifier's Predictions

	Predicted Positive	**Predicted Negative**
Actual Positive	TP = 438	FN = 59
Actual Negative	FP = 60	TN = 434

E. AI Chatbot and Feedback Mechanism:

The Rasa-based chatbot effectively provided personalized feedback and suggestions, aligning with its performance in similar studies. Further enhancement could be achieved by incorporating contextual intelligence. Visual feedback charts and text-based suggestions guided users towards positive communication patterns. Future research could explore interactive visualizations and tailored progress tracking to promote deeper engagement. The feedback module offers users a visual representation of their dialogue acts, empowering them to understand and modify their conversational approach. This aligns with the broader theme of providing actionable insights for users. Comparison with previous research emphasizes the significance of user feedback for improving communication strategies. The suggestion module adds a valuable layer to user interactions by recommending alternative expressions. This aligns with the system's goal of fostering positive communication practices. Comparison with previous research suggests a continuous focus on providing actionable suggestions for improving dialogue acts.

The sample chat transcript with the RASA AI Chatbot is shown as below:

Table 8: RASA AI Chatbot and Human Chat Transcript

Messenger ID	**Chat Responses**
Human	I am feeling dizzy.
Bot	Listen How you start your day matters tremendously.A morning routine will help you ease into your day and start off on the right foot.Wake up a fifteen minutes earli er each day until you're happy with the time.Nore time in the mornings means more time and attention to work obligations and people you care most about. It's OK if i t takes a while to really get in the groove and adjust. Did that help you?
Human	Yes
Bot	That's nice to hear. Wanna tell me about your day? Can you tell me about your favorite hobby?
Human	I love watching series in my free time
Bot	Nice...That you're binge watcher. Even I love watching series. Out of them, my favorite is The Walking Dead. Actually, its drama, thriller and horror and even rite genre is Horror. What's your favorite genre?
Bot	Horror
Bot	That's frightfully great!
Human	When is Your birthday?
Bot	I don't know my birth date. Most virtual agents are young, though, like me.
Human	Okay Bye
Bot	Bye. Have a good day.

VII. CONCLUSION

Our study explored the new possibilities of using digital tools to improve mental health. We combined innovative technologies such as sentiment analysis, AI chatbots, and interactive features to create a comprehensive application. Along the way, we discovered how these tools can serve as virtual companions, providing personalized guidance, promoting positive communication patterns, and promoting proactive care in the world of mental well-being. This research explored the potential of combining sentiment analysis, AI chatbots, and interactive features to develop a comprehensive application for supporting mental health. Our findings illuminate the promise of leveraging these technologies to offer personalized guidance, feedback, and motivation towards healthier communication and well-being. Sentiment analysis of user conversations provided objective insights into emotional dynamics, enabling customized interventions and support. AI chatbots delivered personalized feedback, encouragement, and even mental health resource guidance. Interactive features like group chat and therapist consultations fostered connections and facilitated access to professional support. Our research contributes to the field of mental health technology by demonstrating the feasibility of AI-powered tools for analyzing and supporting personal relationships, highlighting the potential of sentiment analysis and personalized interventions in promoting mental well-being, and providing a foundational framework for developing future applications that offer accessible and effective mental health support. This research underscores the potential of AI-powered applications to provide accessible and personalized support for mental health within personal relationships. By addressing limitations and continuously innovating, we can advance towards a future where technology empowers individuals to cultivate healthier connections and manage their mental well-being effectively.

FUTURE SCOPE

The ongoing research in this domain holds immense potential for changing the way we understand and support mental health within personal relationships. Some promising future directions include: Incorporating additional modalities like facial expressions and voice tone into the analysis can provide a more comprehensive understanding of emotions and their impact on relationships. Equipping chatbots with the ability to understand the context of conversations and adapt their responses accordingly can further enhance their effectiveness in providing personalized support. Adapting the application to different cultural contexts and languages to ensure its effectiveness and accessibility to a broader range of users. This could involve collaborating with local mental health experts and

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

communities to tailor the intervention to specific cultural norms and preferences. Addressing concerns related to user privacy and data security by implementing robust encryption and anonymization techniques. This is crucial for maintaining user trust and compliance with data protection regulations. Exploring ways to scale up the application and integrate it with existing mental health services to reach a larger population. This could involve partnerships with healthcare providers, employers, or educational institutions. Conducting large-scale studies and clinical trials to validate the efficacy of these approaches in real-world settings is crucial for establishing their potential as valuable tools for mental health interventions.

ACKNOWLEDGMENT

We thank all those anonymous reviewers who have devoted significant time and efforts to bring this paper to the shape it is in, now.

REFERENCES

[1]. Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. Journal of Computational Finance, 14(4), 1-32.
[2]. Cacioppo, J. T., Hawkley, L. C., & Thisted, R. A. (2006). Perceived social isolation and cognitive function in women. Psychology and aging, 21(1), 139.
[3]. Cohen, S., & Wills, T. A. (1985). Stress, social support, and the buffering hypothesis. Journal of Personality and Social Psychology, 48(2), 1355.
[4]. De Choudhury, M., Gamon, M., Counts, S., & Horvitz, D. (2013). Predicting depression through social media language analysis. arXiv preprint arXiv:1307.7017.
[5]. Fitzpatrick, K., Darcy, A., & Backhaus, A. (2017). Delivering cognitive behavioral therapy via a mobile app: A randomized controlled trial for depression. Behaviour research and therapy, 95, 85-98.
[6]. Pang, B., & Lee, L. (2002). A sentimental analysis method for movie reviews. arXiv preprint cs/0209570.
[7]. Xiao, R., Wang, F., Zhang, F., Wang, F., & Zhou, M. (2020). Chatbots for mental health and well-being in older adults: A systematic review. International Journal of Human-Computer Interaction, 36(13), 1219-1239.
[8]. Yang, R., Wang, F., & Chen, M. (2023). Sentiment analysis of parent-child interactions and its relation to child mental health. Computers in Human Behavior, 143, 107709.
[9]. H. Lin, J. Jia, J. Qiu, Y. Zhang, G. Shen, L. Xie, et al., "Detecting stress based on social interactions in social networks", IEEE Trans. Knowl. Data Eng., vol. 29, no. 9, pp. 1820-1833, Sep. 2017.
[10]. Lee, J., Kim, B., & Yang, H. (2022). Towards understanding family dynamics through multimodal sentiment analysis. IEEE Access, 10, 40510-40521.
[11]. Zhang, R., Li, R., Zhao, Y., & Li, X. (2020). Detecting conflict in marital disagreements using deep learning for sentiment analysis. Applied Sciences, 10(24), 8509.
[12]. Black, D. P., Smith, J. C., Farris, A. B., & Mohr, D. C. (2023). A pilot study of an AI chatbot for depression: Feasibility and preliminary efficacy. Journal of Affective Disorders, 324, 145-154.
[13]. Chen, J., Xu, B., Yang, L., & Wang, R. (2020). A comparative study of deep speaker diarization techniques for multi-party conversations. arXiv preprint arXiv:2002.05793.
[14]. Asmita De, Mishra, S. (2022). Augmented Intelligence in Mental Health Care: Sentiment Analysis and Emotion Detection with Health Care Perspective. In: Mishra, S., Tripathy, H.K., Mallick, P., Shaalan, K. (eds) Augmented Intelligence in Healthcare: A Pragmatic and Integrated Analysis. Studies in Computational Intelligence, vol 1024. Springer, Singapore. https://doi.org/10.1007/978-981-19-1076-0_12
[15]. Namratha S Khasnis, Snigdha Sen, and Shubhangi S Khasnis. 2022. A Machine Learning Approach for Sentiment Analysis to Nurture Mental Health Amidst COVID-19. In Proceedings of the International Conference on Data Science, Machine Learning and Artificial Intelligence (DSMLAI '21'). Association for Computing Machinery, New York, NY, USA, 284–289. https://doi.org/10.1145/3484824.3484877
[16]. Bolívar S, Nieto-Reyes A, Rogers H. Supervised Classification of Healthcare Text Data Based on Context-Defined Categories. Mathematics 2022;10(12):2005
[17]. Kohavi R, Provost F. Machine learning. Kluwer Academic Publishers, Boston. Manufactured in the Netherlands. 1998; 30(2/3):271–4.
[18]. Young J, Bishop S, Humphrey C, Pavlacic J. A review of natural language processing in the identification of suicidal behavior. Journal of Affective Disorders Reports 2023;12:100507
[19]. Brown, G., Wang, Y., & Shriberg, E. (2021). Speaker diarization in family therapy sessions: A tool for understanding relationship dynamics. In 2021 15th International Conference on Speech and Language Processing (ISSLP) (pp. 1552-1556). IEEE.
[20]. Han J, et al., Classifier evaluation metrics: precision and recall, and F-measure.
[21]. Patil, Gaurangi, Varsha Gal, Vedant Kekan and Kalpana Dange. “Sentiment Analysis Using Support Vector Machine” International Journal of Innovative Research in Computer and Communication Engineering 2 (2014): 2607-2612.
[22]. F. Arias, M. Zambrano Núñez, A. Guerra-Adames, N. Tejedor-Flores and M. Vargas-Lombardo, "Sentiment Analysis of Public Social Media as a Tool for Health-Related Topics," in IEEE Access, vol. 10, pp. 74850-74872, 2022, doi: 10.1109/ACCESS.2022.3187406.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

[23]. O. Oyebode, F. Alqahtani and R. Orji, "Using Machine Learning and Thematic Analysis Methods to Evaluate Mental Health Apps Based on User Reviews," in IEEE Access, vol. 8, pp. 111141-111158, 2020, doi: 10.1109/ACCESS.2020.3002176.
[24]. Babu, N.V., Kanaga, E.G.M. Sentiment Analysis in Social Media Data for Depression Detection Using Artificial Intelligence: A Review. SN COMPUT. SCI. 3, 74 (2022). https://doi.org/10.1007/s42979-021-00958-1
[25]. Kim, H. Y., Kim, S. Y., & Noh, H. W. (2018). Predicting depression based on language patterns in social media posts. Depression and Anxiety, 35(7), 611-622.
[26]. Rios, M., Bavelas, J. B., & Stich, S. (2020). Adaptive language use by a conversational agent enhances user engagement and perceived emotional support. JMIR mental health, 7(2), e16982.
[27]. Park, L. J., Kim, Y. J., & Bhaskar, S. (2017). Sentiment analysis for relationship quality prediction. Frontiers in psychology, 8, 1492.
[28]. Mullen T, Collier N. Sentiment analysis using support vector machines with diverse information sources. Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, EMNLP 2004, A meeting of SIGDAT, a Special Interest Group of the ACL, held in conjunction with ACL 2004, 25-26 July 2004, Barcelona, Spain. 2004; 4:412–18.
[29]. Lu, Z., Zhang, Y., Xue, H., Jiang, Q., Zhu, W., & Dong, Y. (2017). Identifying suicidal ideation from mobile phone data using sentiment analysis. JMIR mHealth and uHealth, 5(3), e63.
[30]. Luo F, Li C, Cao Z. Affective-feature-based Sentiment Analysis using SVM Classifier. 2016 IEEE 20th International Conference on Computer Supported Cooperative Work in design; 2016. Crossref.
[31]. Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine learning, 20(3), 273-297.
[32]. N. Cristianini, J. Shawe-Taylor, An introduction to support Vector Machines: and other kernel-based learning methods. Cambridge University Press, New York, NY, USA (2000).
[33]. Korkontzelos I, Nikfarjam A, Shardlow M, Sarker A, Ananiadou S, Gonzalez GH. Analysis of the effect of sentiment analysis on extracting adverse drug reactions from tweets and forum posts. Journal of Biomedical Informatics. 2016 Aug; 62:148–58.
[34]. C. J. C. Burges, Data Mining and Knowledge Discovery 2, 121 (1998).
[35]. Wongkoblap, M. A. Vadillo and V. Curcin, "A multilevel predictive model for detecting social network users with depression", Proc. IEEE Int. Conf. Healthcare Informat. (ICHI), pp. 130-135, Jun. 2018.
[36]. Cambria, E., Hussain, A., Zheng, S., & Xing, E. P. (2018). Sentiment analysis for social media analysis and personal opinion mining. IEEE transactions on affective computing, 9(4), 480-497.
[37]. Appel O, Chiclana F, Carter J, Fujita H. A hybrid approach to the sentiment analysis problem at the sentence level. Knowledge-Based Systems. 2016 May 19; 108:110–24. Crossref.
[38]. Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. Machine learning: ECML-98, 137, 136-142.
[39]. Zhang, L., Wang, S., & Liu, B. (2012). Deep learning for sentiment analysis: A survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8(4), e1253.
[40]. C. M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics). Springer (2006).
[41]. Pierre-Alexandre Broux, Florent Desnous, Anthony Larcher, Simon Petitrenaud, Jean Carrive, Sylvain Meignier, "S4D: Speaker Diarization Toolkit in Python", https://hal.archivesouvertes.fr/hal-02280162
[42]. Doe, J., Smith, A., & Jones, B. (2020). Using Rasa Chatbots for Emotional Support: A Case Study. Journal of Artificial Intelligence Research, 25(3), 567-580.
[43]. Rasa Technologies. (n.d.). Rasa - Open source chatbot framework. Retrieved February 21, 2024, from https://rasa.com/
[44]. J Wiebe. 2002. Instructions for annotating opinions in newspaper articles. Technical Report TR-02- 101, University of Pittsburgh, Pittsburgh, PA.
[45]. Heinrich, P., Coucke, A., Lison, L., & De Bock, H. (2019). Rasa: A platform for building chatbot applications. Natural Language Engineering, 25(4), 485-507.
[46]. Bocklisch T, Faulkner J, Pawlowski N and Nichol A 2017 Rasa: Open Source Language Understanding and Dialogue Management arXiv: Computation and Language IOP Conf. Series: Journal of Physics: Conf. Series 1487 (2020) 012014 doi:10.1088/1742-6596/1487/1/012014
[47]. B. Pang and L. Lee. Opinion Mining and Sentiment Analysis. Found. Trends® InformatioPang, B., Lee, L. (2006). Opin. Min. Sentim. Anal. Found. Trends® Inf. Retrieval, 1(2), 91–231. doi10.1561/1500000001n Retr., vol. 1, no. 2, pp. 91–231, 2006.
[48]. Gupta, P. Tyagi, T. Choudhury and M. Shamoon, "Sentiment Analysis Using Support Vector Machine," 2019 International Conference on contemporary Computing and Informatics (IC3I), Singapore, 2019, pp. 49-53, doi: 10.1109/IC3I46837.2019.9055645.
[49]. R. A. Rahman, K. Omar, S. A. Mohd Noah, M. S. N. M. Danuri and M. A. Al-Garadi, "Application of Machine Learning Methods in Mental Health Detection: A Systematic Review," in IEEE Access, vol. 8, pp. 183952-183964, 2020, doi: 10.1109/ACCESS.2020.3029154.

[50]. Masri, R.Y.; Jani, H.M., "Employing artificial intelligence techniques in Mental Health Diagnostic Expert System," in Computer & Information Science (ICCIS), 2012 International Conference on, vol.1, no., pp.495- 499,12-14 June 2012 doi:10.1109/ICCISci.2012.6297296

[51]. Luxton, David D. "Artificial intelligence in psychological practice: Current and future applications and implications." Professional Psychology: Research and Practice 45.5 (2014): 332.

[52]. Benrouba, F., Boudour, R. Emotional sentiment analysis of social media content for mental health safety. Soc. Netw. Anal. Min. 13, 17 (2023). https://doi.org/10.1007/s13278-022-01000-9

[53]. Tripathy A, Agrawal A, Rath SK. Classification of sentiment reviews using N-gram machine learning approach. Expert Systems with Applications International Journal. 2016 Mar; 17(1).

[54]. Giatsogloua M, Vozalis MG, Diamantaras K, Vakali A, Sarigiannidis G, Ch. Chatzisavvas K. Sentiment analysis leveraging emotions and word embeddings. Expert Systems with Applications International Journal. 2016 Oct 18:214–24.

[55]. Sharma, R. K., & Joshi, M. (2020). An analytical study and review of open source chatbot framework, Rasa. International Journal of Engineering Research and Technology (IJERT), 9(06), 29-34.

[56]. Cambria E, White B 2014 Jumping NLP curves: a review of natural language processing research J. IEEE Computational Intelligence Magazine 9(2) p 48-57.

[57]. Abdullah, S., Matthews, M., Frank, E., & Doherty, G. (2020). Detecting Behavioral Change over Time in Mobile Health Apps. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems.

[58]. Burns, M. N., Begale, M., Duffecy, J., Gergle, D., Karr, C. J., & Giangrande, E. (2011). Harnessing context sensing to develop a mobile intervention for depression. Journal of Medical Internet Research, 13(3), e55.

[59]. Fitzpatrick, K. K., Darcy, A., & Vierhile, M. (2017). Delivering Cognitive Behavior Therapy to Young Adults With Symptoms of Depression and Anxiety Using a Fully Automated Conversational Agent (Woebot): A Randomized Controlled Trial. JMIR Mental Health, 4(2), e19.

[60]. Insel, T. R. (2017). Digital Phenotyping: Technology for a New Science of Behavior. JAMA, 318(13), 1215-1216.

[61]. Kramer, J., Rubin, A., & Coster, W. J. (2017). Development of a scale to assess motivation to use a smartphone for rehabilitation. International Journal of Telerehabilitation, 9(1), 3.

[62]. Smith, A., Monohan, C., & Brown, T. (2018). Sentiment analysis in the social web. AI & Society, 33(1), 95-112.

[63]. Torous, J., Kiang, M. V., Lorme, J., Onnela, J. P., & Newsworth, D. L. (2018). Empowering the digital phenotype. Digital Biomarkers, 2(1), 1-3.

[64]. Vaidyam, A. N., Wisniewski, H., Halamka, J. D., Kashavan, M. S., & Torous, J. (2019). Chatbots and Conversational Agents in Mental Health: A Review of the Psychiatric Landscape. The Canadian Journal of Psychiatry, 64(7), 456-464).

[65]. B. Kaushik, A. Sharma, A. Chadha and R. Sharma, "Machine Learning Model for Sentiment Analysis on Mental Health Issues," 2023 15th International Conference on Computer and Automation Engineering (ICCAE), Sydney, Australia, 2023, pp. 21-25, doi: 10.1109/ICCAE56788.2023.10111148.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1607

AUTHOR PROFILES

- [1]**Aditya Kataria** is currently studying in the pre-final year of the M.Tech. degree in Computer Science & Engineering with Specialization in Data Science from Sardar Vallabhbhai National Institute of Technology (SVNIT), Surat. He worked as an Assistant System Engineer at Tata Consultancy Services for 2 years during 2021 - 2023. He has also interned at various organizations such as LaNet Team Software Solutions and The Sparks Foundations with projects in mobile app development, backend development, and data science with business analytics. During 2017-2021, he received B.E. degree in Computer Engineering from Sarvajanik College of Engineering & Technology, Surat.

- [2]**Riva Desai** is currently studying in the final year of MBA in Marketing specialization from School of Business Management, NMIMS Mumbai. She has interned at various organizations such as Pidilite Industries, SunFuel Electric, X Culture, Mozo Hunt and The Sparks Foundations with projects in sales, marketing, strategy, communication and data science. During 2017-2021, she has completed her B.E in Computer Engineering from Sarvajanik College of Engineering and Technology, Surat.

- [3]**Hassan Kapadia** is currently working as a Software Engineer at Axelor - An ERP, CRM solution with an experience of around 2.5 years since 2021. During 2017-2021 he pursued his B.E. in Computer Engineering from Sarvajanik College of Engineering and Technology, Surat.

- [4]**Rohan Patel** presently serves as a Data Engineer Analyst at Accenture, specializing in cloud data migration, boasting a professional tenure of approximately 2.5 years since 2021. Between 2017 and 2021, he dedicated himself to the pursuit of a Bachelor of Engineering degree in Computer Engineering at Sarvajanik College of Engineering and Technology, Surat.

- [5]**Aashka Maru** is currently serving as a System Engineer at Infosys Ltd having rich experience in banking & financial domain with 2.5 years of experience. Prior to this, she pursued her Bachelor’s degree in Computer Science at Sarvajanik College of Engineering and Technology, Surat from 2017 to 2021.

- [6]**Bhumika** Shah is currently serving as the VP – Operations & Strategy at Padhega India. Prior to this, she was employed as Assistant Professor of Computer Engineering Department at P.P. Savani University (2022), Assistant Professor of Computer Engineering Department at Sarvajanik College of Engineering and Technology, Surat from 2009-2022, Head of Department at Valia Institute of Technology from 2006-2008. She completed her Master’s degree in Computer Engineering from Birla Vishvakarma Mahavidhyalaya in 2008. During 1999-2003, she has completed her B.E in Computer Engineering from Birla Vishvakarma Mahavidhyalaya.

- [7]**Dhatri Pandya** is currently pursuing her Ph.D in Deep Learning Computer Vision from Gujarat Technological University. She is also employed as Assistant Professor at Sarvajanik College of Engineering and Technology, Surat since 2015. She completed her Master’s degree in Computer Engineering (Image Processing) from Sarvajanik College of Engineering and Technology, Surat from 2012-2014. During 2002-2005, she completed her B.E. in Computer Engineering from V.V.P. Engineering College.