T.C.
ISTANBUL GEDIK UNIVERSITY
INSTITUTE OF GRADUATE STUDIES

THE IMPACT OF STRESS ON DIABETES MANAGEMENT – A SENTIMENT ANALYSIS APPROACH

MASTER'S THESIS

Engy Mohamed Khalil ALI

Department of Data Science

Master's Program in Statistics and Data Science (Thesis) in English

AUGUST 2025
ISTANBUL

T.C.
ISTANBUL GEDIK UNIVERSITY
INSTITUTE OF GRADUATE STUDIES

THE IMPACT OF STRESS ON DIABETES MANAGEMENT – A SENTIMENT ANALYSIS APPROACH

MASTER'S THESIS

Engy Mohamed Khalil ALI
(231225004)

Department of Data Science

Master's Program in Statistics and Data Science (Thesis) in English

Thesis Advisor: Asst. Prof. Halime SUVAY EKER

Istanbul 2025

T.C.
İSTANBUL GEDİK ÜNİVERSİTESİ
Lisansüstü Eğitim Enstitüsü Müdürlüğü

Jüri Tez Onay Formu

21.08.2025

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ MÜDÜRLÜĞÜ

Bu çalışma 21.08.2025 tarihinde aşağıdaki jüri tarafından Veri Bilimi Anabilim Dalı, İstatistik ve Veri Bilimi İngilizce (Tezli Yüksek Lisans) Programı Yüksek Lisans Tezi olarak kabul edilmiştir.

TEZ JÜRİSİ

Dr. Öğr. Üyesi Halime SUVAY EKER
Danışman
İstanbul Gedik Üniversitesi

Üye (İmza)
Dr. Öğr. Üyesi Erdoğan BOZKURT
İstanbul Gedik Üniversitesi

Üye (İmza)
Dr. Öğr. Üyesi Duygu DEMİRAY AKKAYA
İstanbul Topkapı Üniversitesi

DECLARATION

I Engy Mohamed Khalil ALI hereby declare on my honor that the Master's thesis titled " The Impact of Stress on Diabetes Management – A Sentiment Analysis Approach" has been written without resorting to any assistance that would contravene scientific ethics and traditions throughout all processes from the project phase to its conclusion, and that the works I have utilized are those listed in the Bibliography, and they have been used with proper citation (21/08/2025).

Engy Mohamed Khalil ALI

PREFACE

First of all, I want to thank my academic advisor, Asst. Prof. Halime Suvay EKER, for all the help, support, and encouragement she gave me while I was working on this project. Her advice, suggestions, and knowledge have all been very helpful in writing this thesis.

I also want to thank the University staff and teachers for making it a good place to learn and giving me the tools I needed to finish this study.

I really appreciate how my family and friends are always there for me, love me, and understand me when things get hard on this trip. You believing in me is what has pushed me the most.

Finally, I want to thank the people and groups on the internet whose stories and experiences helped me with this research. It would not have been possible to do this study without their willingness to share and help.

August 2025 Engy Mohamed Khalil ALI

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

DDS	: Diabetes Distress Scale
HbA1c	: Glycolated Hemoglubin
LDA	: Latent Dirichlet Allocation
ML	: Machine Learning
NLP	: Natural language processing
UK	: United Kingdom

LIST OF TABLES

Page No.

LIST OF FIGURES

Page No.

THE IMPACT OF STRESS ON DIABETES MANAGEMENT – A SENTIMENT ANALYSIS APPROACH

ABSTRACT

Managing diabetes is a lifelong process and self-adhering job that requires more than just taking the diabetic medications. It also requires extra care for mental and emotional health. Stress is one of the most important mental factors that makes it hard to manage diabetes well right now. This study looks at how stress affects diabetes management by looking at patient stories from online health forums and using sentiment analysis. The study uses natural language processing (NLP) tools like TextBlob and VADER to look at 792 patients' comments on online forums, also topic modeling analysis to check the dominant topics among the comments. It looks for patterns of negative emotions and feelings that people have when they are stressed in the comments. And to see how stress affects managing diabetes as an emotional and mental factor.

The overall results findings show that stress has a big effect on how people with diabetes feel when they tell their stories, like the ones they post on online forums. It has been observed that many comments are negative and contain highly subjective feelings, particularly regarding mental health issues, treatment compliance and lifestyle. But we can't say for sure that stress always makes all diabetics act out in a bad way. Some themes were closely related to negative emotional intensity and emotional burden, while other themes were more closely related to coping, resilience, and positive support.

The most discussed topic was Blood sugar control, then Blood sugar &stress with difference in 10 comments between them, finally stress management topic . Sentiment analysis indicated that there was much emotion regardless of the neutral tone. The tri-topic model coherence increased from 0.099 and 0.329 to 0.430. "Blood Sugar Control" carried the highest VADER score (0.430), indicating strong emotional expression and a greater need for support in managing glucose levels. In contrast, "Blood Sugar and Stress" had the lowest sentiment score (0.099), suggesting a more neutral or less emotionally expressive tone in this topic.

Keywords: *Diabetes; Stress; Sentiment analysis; Machine learning; Patients narratives*

STRESİN DİYABET YÖNETİMİ ÜZERİNDEKİ ETKİSİ – BİR DUYGU ANALİZİ YAKLAŞIMI

ÖZET

Diyabet yönetimi, sadece diyabet ilaçlarının alınmasından daha fazlasını gerektiren, yaşam boyu devam eden ve kişinin kendine bağlılık gösterdiği bir süreçtir. Bu süreçte, zihinsel ve duygusal sağlığa yönelik ekstra özen göstermek de önem taşımaktadır. Stres, diyabetin etkin bir şekilde yönetilmesini zorlaştıran en önemli zihinsel faktörlerden biridir. Bu çalışma, çevrimiçi sağlık forumlarındaki hasta paylaşımlarını inceleyerek ve duygu analizi yöntemlerini kullanarak stresin diyabet yönetimi üzerindeki etkisini araştırmaktadır. Çalışmada, TextBlob ve VADER gibi doğal dil işleme (NLP) araçlarıyla 792 hastanın forumlardaki yorumları analiz edilmiştir. Yorumlarda, stres altında ortaya çıkan olumsuz duygu ve his kalıpları tespit edilerek, stresin duygusal ve zihinsel bir faktör olarak diyabet yönetimine etkisi incelenmiştir.

Genel bulgular, stresin diyabetli bireylerin deneyimlerini anlatırken özellikle çevrimiçi forumlarda paylaştıkları hikayelerde duygusal durumları üzerinde büyük bir etkisi olduğunu göstermektedir. Özellikle zihinsel sağlık sorunları, tedaviye uyum ve yaşam tarzı konularında birçok yorumun olumsuz ve oldukça öznel duygular içerdiği tespit edilmiştir. Ancak stresin her zaman tüm diyabetlilerin olumsuz davranışlar sergilemesine yol açtığını kesin olarak söylemek mümkün değildir. Bazı temalar, olumsuz duygu yoğunluğu ve duygusal yükle yakından ilişkili bulunurken, diğer temalar ise başa çıkma, dayanıklılık ve olumlu destekle daha fazla bağlantılı bulunmuştur.

En çok tartışılan konu, yiyecek ve ilaçla ilgili stres olmuştur. Duygu analizi, nötr üsluba rağmen çok fazla duygu olduğunu ortaya koymuştur. Üçlü konu modeli uyumluluğu 0.099 ve 0.329değerlerinden 0.430'ya yükselmiştir. "Kan Şekeri ve Stres" konusu, VADER skorunun en yüksek olduğu alan olup (0.430), bu durum duygusal yoğunluğun yüksek olduğunu ve diyabetliler için önemli derecede destek ihtiyacının bulunduğunu göstermektedir.

Anahtar kelimeler: *Diyabet; stress; Duygu analizi; Makine öğrenimi; Hasta deneyimleri*

1. INTRODUCTION

1.1 Subject of the Study

Diabetes mellitus is a chronic metabolic disorder that leads to elevated blood glucose levels, either due to the inability of the body to produce sufficient insulin or due to the lack of proper functioning insulin produced both occurred (Zarei, 2021).

Insulin is the hormone of the pancreas and plays a role in the way in which it reacts to the breakdown of glucose. Hyperglycemia is a state where the body does not make sufficient insulin or respond improperly to insulin. This causes improper buildup of glucose way up in the blood stream. Diabetes is now a priority public health problem globally, and its prevalence rates are single-mindedly increasing. Some of the reasons this has been the case are that older people have not been exercising as much and having more unhealthy diets (American Diabetes Association, 2023)

Diabetes can lead to serious long-term complications like heart disease, neuropathy, nephropathy, retinopathy, and limb amputations if it is poorly controlled (INTERNATIONAL DIABETES FEDRETION,, 2022). There are various categories of diabetes, and each of them has its causes and cure processes. Type 1 diabetes is an autoimmune condition in which the immune system of the body targets the beta cells of the pancreas that are insulin-releasing cells

Due to this, individuals with type 1 diabetes require insulin treatment for the rest of their lives. Type 2 diabetes, however, is primarily brought on by insulin resistance and typically associated with being overweight, physical inactivity, and a poor diet. Gestational diabetes occurs during pregnancy and typically resolves after giving birth. Monogenic and secondary diabetes are two of the rarer ones. They may be due to other diseases or due to certain drugs. Diabetes affects not just the internal organs, but mental and emotional status as well. Monitoring your blood sugar levels daily, adhering to your medications, and modifying your lifestyle can all have

adverse effects on emotional stress. Diabetes distress is a useful theory in this context.

It indicates the psychological disturbance of coping with the illness. Although different from clinical depression, they happen at the same time and have more negative consequences to health (Fisher et al, 2012) Diabetics are also more likely to be depressed and anxious. These states render people less active, render them less capable of taking care of themselves, and render them unable to control their blood sugar levels (Gonzalez et al, 2008)

Individuals with diabetes are also more likely to be anxious and depressed. These conditions can make it more difficult for people to be motivated, for them to manage to look after themselves, and for them to manage their blood sugars (Gonzalez et al, 2008).

Stress can increase the levels of cortisol, which can cause the body to strain more when it comes to regulating glucose and exacerbating insulin resistance (Cohen et al, 2019). Stress also has deleterious impacts on significant health behaviors such as diet, physical activity, and sleep quality.

An individual's emotional state is likely to determine how well they can adhere to their treatment, follow their blood sugar, and become involved in decisions regarding their health care. while individuals with high emotional resilience and high levels of social support are likely to get better and feel better (Chew et al, 2016).

Since the comments came from https://www.diabetes.co.uk/forum/, it's important to show the rate of diabetes in the UK. In 2025, diabetes is still a major public health issue in UK, with an estimated 5.9 million adults living with the condition, both diagnosed and undiagnosed (World PopulationReview, 2025). The latest UK Government diabetes profile reports that approximately 7.0% of people over the age of 17 have type 2 diabetes, according to general practice ([OHID], 2025). Diabetes UK also estimate that 1.3 million remain undiagnosed. But add to the 6.3 million who are pre-diabetic, one in every five adults in the UK are either diabetic or at risk (Marsh, 2025). This is due to the fact that emotional stress has a direct correlation with negative diabetes self-management outcomes. Diabetes is also very costly to the healthcare system; almost one-third of cardiovascular disease deaths in the UK occur in individuals with diabetes (Campbell, 2025).

1.2 Purpose of the Study

This research aims to bridge that gap by investigating how individuals with diabetes discuss and manage stress and the way it impacts their diabetes care through online health forums. These individual stories are analyzed using sentiment analysis and natural language processing (NLP) like TextBlob and VADER to identify prevalent emotional patterns and themes concerning stress.

1.3 Hypothesis

Four general hypotheses inform the research:

- (H1) Stress leaves a quantifiable negative mark on the emotional tone of fiction written by individuals with diabetes.
- (H2) Tales discussing problems regarding stress are associated with indicators of inadequate self-management, such as missing medication and having problems with food.
- (H3) Support networks that may be used in an attempt to offset adverse stress-influenced thoughts and emotions include doctors, family members, and online forums.
- (H4) Aspect-based sentiment analysis captures emotional fluctuation by topic.

Collectively, these findings support the hypothesis that stress differentially affects diabetes management domains, with physiological controls prioritized over emotional coping. Future work may leverage domain-adapted transformers to enhance negative-sentiment detection and explore interventions that integrate stress-management education into diabetes care.

The main findings indicate that stress has a central role to play when people with diabetes feel as they narrate their stories, such as the ones they write on online forums. The responses were seen to be largely negative and transmit highly subjective feelings, especially regarding mental illness, adherence to therapy and lifestyle. But we can't be certain that stress always causes all diabetics to behave negatively. There were some of the themes that were more clearly associated with bad emotional intensity and emotional burden, and there were some other themes that were more clearly associated with coping, resilience, and good support.

2. LITERATURE REVIEW

The chronic nature of diabetes demands ongoing behavioral together with psychological and physiological assessment for management purposes. Research shows stress functions as an important psychological obstacle which affects disease outcomes among different elements. Stress interferes with blood sugar regulation by releasing cortisol during physical responses and through behavioral responses such as medication and self-care avoidance (al B. e., 2010-2022)

Patient narratives on the internet give researchers valuable insights into diabetes management experiences while allowing analysis through Sentiment analysis which is a Natural Language Processing (NLP) technique that evaluates emotional patterns in extensive patient-generated text.

Stress that persists in the long term causes negative effects on diabetes control together with disease progression. The researchers Fisher et al. (2012) identified diabetes distress as a complex emotional state which emerges from managing chronic illness while leading to worse treatment adherence along with unstable glycemic control. The research conducted by Lloyd et al. (2019) together with Hackett & Steptoe (2017) established that psychological stress creates a direct link to elevated insulin resistance and elevated HbA1c levels.

Stress reduction therapies combined with counseling have produced promising effects on clinical results as well as stress levels among patients with diabetes (Wang et al, 2020) .The majority of research has used clinical data but there is a scarcity of investigations about stress expressions that appear in everyday language within digital health narratives.

The emotional state of diabetes-related distress creates major obstacles for patients to perform self-care activities and maintain their motivation. The Diabetes Distress Scale (DDS) developed by Fisher et al. (2012) measured emotional distress and revealed that new diabetes patients reported high distress levels. Reddy & Shankar Palli (2021) demonstrated through their research that negative emotional

statements in online diabetes support groups create problems with medication compliance and intensify psychological distress.

The research conducted by (Kuru & Akbulut , 2021) together with Islam et al. (2018) proved that deep learning models including BERT and Naïve Bayes along with LSTM deliver effective sentiment classification results. The research conducted by Johnson & Lee (2021) proved that Twitter emotional patterns serve as predictive indicators of mental health states.

Sentiment analysis emerges as a beneficial instrument which reveals concealed emotional patterns in patient communication according to these research findings. The application of these tools to stress-related discussions in diabetes still presents an unaddressed opportunity for research.

(Zhang, 2020)examined forum comments to find that stress triggers emerged from medication-related frustrations and dietary challenges and insufficient social support. Brown et al. (2022) integrated topic modeling with sentiment analysis to study diabetes patients' emotional states which demonstrates the value of digital narratives for understanding disease burden.

The existing research has limitations because these approaches fail to identify specific stress-related expressions which hinders the development of targeted interventions for psychological distress reduction.

The processing of unstructured health data through Natural Language Processing has enabled researchers to extract significant meaningful insights. The application of NLP in diabetes research involves three main applications including emotional theme detection and concern classification and patient mood classification (Smith et al, 2020). NLP systems encounter two main drawbacks because they need sentiment-labeled datasets and they struggle with processing the informal text found in social media and online forums.

The sentiment scoring tools TextBlob and VADER offer easy-to-use results but transformer models like BERT deliver more precise language pattern detection when trained with domain-specific corpora.

Research conducted by (Cohen et al. and Miller et, 2019-2020) demonstrates that high stress levels directly correlate with worse glycemic control. High levels of stress lead to both higher A1C readings and lower self-monitoring frequencies and

higher rates of medication noncompliance. Real-world healthcare centers face challenges when adopting behavioral stress management programs because these programs demonstrate success in treating both mental health and physical health issues. The application of sentiment analysis in healthcare exists but most research studies analyze overall emotional expressions and disease impacts instead of focusing on stress-related stories. The research field shows a significant shortage of studies which use NLP tools to analyze genuine unstructured patient feedback about stress in diabetes management.

The research shows stress plays an essential role in how people manage their diabetes. Sentiment analysis provides strong methods to study healthcare emotions yet researchers need to develop better approaches to study diabetes-related stress. The research examines this knowledge gap by using sentiment analysis tools to evaluate stress-related themes in patient-generated text from online forums and social media platforms. The research produces two important outcomes by creating monitoring systems and emotional support strategies for diabetic patients who experience stress.

Table 1: Previous Studies on Stress, Diabetes, and Sentiment Analysis

Study (APA Style)	Methodology / Algorithm	Dataset / Source	Findings / Accuracy
Fisher, L., et al. (2012)	Psychometric scales	Clinical interviews with diabetic patients	Defined “diabetes distress” as a distinct emotional burden
Gonzalez, J. S., et al. (2008)	Meta-analysis	42 studies on depression in diabetes	Depression associated with poor glycemic control
Chew, B. H., et al. (2016)	Survey-based stress analysis	356 T2D patients in Malaysia	High stress linked with poor self-care; support improved outcomes
Cohen, S., et al. (2019)	Biological stress marker correlation	Clinical cortisol testing in diabetic patients	Stress worsens insulin resistance via cortisol
Zhou, Y., et al. (2021)	Sentiment Analysis (VADER, LIWC)	Reddit diabetes forums	Negative emotion more prevalent in posts about stress & burnout
Liu, X., et al. (2020)	Topic Modeling + Sentiment Analysis	Chinese health forums	Identified stress and fear as dominant negative themes
Wang, H., et al. (2022)	BERT-based sentiment classification	Patient reviews from online diabetes communities	85% accuracy in detecting emotional polarity

Zarei, F., et al. (2021)	Literature review	Meta-review of stress-diabetes relationship	Emphasized the need for emotional support in diabetes treatment

3. METHODOLOGY

3.1 Research Design

This research employs a mixed-methods strategy that incorporates qualitative and quantitative analysis to investigate psychological stress effect on diabetes management according to patient feedback posted in online forums. The qualitative method utilizes topic modeling to capture themes in text, and the quantitative method employs software for sentiment analysis to detect emotional tone.

The research combines natural language processing (NLP) and machine learning to extract insights from unstructured data. Through the combination of sentiment scoring and thematic analysis, the research strives to identify the emotional cues and behavioral responses for stress in diabetic patients.

3.2 Purpose and Importance of the Study

The aim of this study is to explore the psychological and emotional effect of stress on diabetic patients, based on real experiences gathered from online discussion forums. Despite as much control of diabetes is achieved clinically and through the mechanism of lifestyle modification, the psychological aspect of stress and emotional distress is insufficiently studied in practice and literature (Chew B. H.-G., 2014) . This study seeks to fill that gap by utilizing Natural Language Processing (NLP) techniques in examining patient-generated material in order to gain insight into how stress is described in language and how it correlates with self-management challenges.

It is worthwhile to study the emotional lives of people with diabetes because uncontrolled stress can disrupt drug compliance, dietary limitation, blood glucose self-regulation, and health status. Using sentiment analysis and topic modeling, this study reveals latent emotional patterns in free text not always evident in clinic situations.

The contribution of this study is its novel direction and its applicability:

• It has a data-driven perspective towards the emotional dimensions of chronic disease living.

• It hears patients too often silenced in clinical data through patient-reported experience focus.

• It emphasizes integrated mental health care inclusion in diabetes disease management programs.

Through filling the gap between computational methods and healthcare understanding, this research enhances understanding of diabetes management and underscores the significance of taking into account mental health as well as physical care.

3.3 Participants / Sampling

The data for this research comes from user comments on two public online forums:

• https://www.diabetes.co.uk/forum/

The users are anonymous individuals on these forums posting about diabetes. A purposive sampling approach was used with keyword filters to identify posts containing both:

• Stress words: "stress", "anxiety", "overwhelmed", "worried", "burnout"

• Diabetes words: "blood sugar", "insulin", "diet", "medication", "CGM"

Only the publicly visible English-language blog posts that did not have a login requirement were used. The original dataset had 792 comments, and after de-duplication and cleaning, there were around 271 useful and distinct posts.

3.4 Data Collection Tool

Python web scraping script was coded and run in the Spyder IDE (in Anaconda environment). The approach utilized the following libraries:

• requests and Beautiful Soup for web scraping

• TRE for regular expression-based text cleaning

• nltk for tokenization, removing stop words, stemming, and lemmatization

• pandas for data handling and storage

Forum thread text data was web scraped and saved in CSV format after preprocessing for suitability of NLP and sentiment analysis. (McKinney, 2017)

3.5 Data Collection Process &Preprocessing

Data collection was accomplished through:

3.5.1. Forum selection

One popular and publicly accessible diabetes forums were selected. Which is https://www.diabetes.co.uk/forum/

3.5.2. Text extraction

Responses and comments were harvested from them via keyword-based filtering using web scraping to collect all the comments

3.5.3 Filtering

Posts were filtered by changing all words to lowercase words, removing all the repetitive words, removing all unnecessary comments, doing lemmatization and tokenization, checking relevance through searching stress and diabetes-related keywords and diabetes, to check for the dominant topics according to the most used and repeated keywords and then starting the sentiment analysis using NLP tools like text blob and Vader.

3.5.4. Cleaning and preprocessing

In natural language processing (NLP) text preprocessing, the starting point is usually to convert all that is in the text to lowercasing and then de-URLing, punctuation removal, emojis removal, and other special characters removal. Tokenization and stop word removal, typically English stop words but potentially including standard stop words as well as custom stop words, would follow. Stemming and lemmatization would follow to further preprocess the text. Stemming is a rule-based approach to reducing a word to its stem or root by stripping off prefixes or suffixes, but not necessarily returning words found in the dictionary. "Running" is reduced to "run" and "happiness" reduces to "happi". One of the most

popular stemming algorithms is the Porter Stemmer developed by Martin Porter in 1980, which uses a set of heuristically defined rules to remove word endings. (Porter, 1980)

Alternatively, lemmatization lowers a word to its lemma or root dictionary word so that output is always a proper word. Lemmatization is typically vocabulary and morphological analysis dependent with part-of-speech tagging sometimes necessary in order to make the proper output. For example, "running" lemmatizes to "run" and "better" to "good". One of the common uses of lemmatization in NLP is the WordNet Lemmatizer based on the WordNet lexical database in retrieving contextual lemmas. (Miller, 1995)

3.5.5. Relevance tagging

The posts were marked as relevant only when they contained stress-related in addition to diabetes-related keywords.

The end result was a clean, high-quality dataset for modeling and analysis.

3.6 Data Analysis

3.6.1 Topic modeling using LDA

Latent Dirichlet Allocation (LDA) is a probabilistic generative model that's widely applied in natural language processing (NLP) to uncover hidden thematic structure in massive text collections. It posits that every document (in this case, a patient review) is a mixture of several topics, and for each topic, there's a word distribution. (Blei, Ng, & Jordan, 2003)

Latent Dirichlet Allocation or LDA is a generative model of probability which represents a document as if it were created via random selection of a topic distribution and subsequently random sampling from words according to these topics.

This model assigns each document a probability distribution of topics, and each topic has a probability distribution of words. The terms in a document are supposed to have been sampled from these distributions. In technical terms, LDA uses Bayesian inference to make an estimation of the hidden topic structure by

approximating two main probabilities: the probability of a topic for a given document, and the probability of a word for a given topic.

Two hyperparameters regulate the sparsity of these distributions: alpha (α), which regulates the document–topic distribution sparsity, and beta (β), which regulates the topic–word distribution sparsity. The model yields two main outputs: the Document–Topic Matrix, showing how frequently each topic occurs in each document, and the Topic–Word Matrix, showing how frequently each word occurs in each topic.

LDA was used in this project as it best operates to uncover underlying emotional and behavioral themes of user-generated content. When applied to diabetes online forums, LDA can classify comments automatically into easily readable topics, uncover relationships between stress and factors like diet, insulin, or work, and create an insightful summary of otherwise unstructured patient narratives. Theme labeling without hand-coding is feasible, as well as being scalable and objective for large data sets. For this study, LDA was run on pre-processed comments. Following tokenization, stemming, and text segmentation, the comments were then transformed into vectors using the Count Vectorizer algorithm. Count Vectorizer is a natural language processing (NLP) text feature extraction algorithm used to convert a list of text documents into a term–document matrix, also known as a document–term matrix. It tokenizes the text, creates a vocabulary of distinct words from the corpus, and determines word frequency for each word in each document. (Harris, 1954)

This is needed because the vast majority of machine learning algorithms expect numeric input, whereas raw text is inherently unstructured. Translating text into numeric feature vectors allows statistical models and algorithms to analyze, process, and learn from text data. It is widely employed as a pre-step in operations like text classification, sentiment analysis, and topic modeling. For the present work, the LDA model was set to generate three topics (components = 3). Each topic was explained using its most common words, and titles were given as Topic 1 – Blood Sugar Control, Topic 2 – Blood Sugar & Stress, and Topic 3 – Stress Management

3.6.2 Sentiment Analysis

Scientific Background:

Sentiment Analysis, or Opinion mining, is the field of NLP which tries to find the emotional polarity of a piece of text. It usually indicates whether the polarity of a statement is positive, negative, or neutral and in some applications, subjectivity (personal opinion vs. fact)

(Liu, 2012)Two established sentiment instruments are used in this study:

A. Text Blob**:** It is a rule-based library for sentiment analysis that stood on the shoulders of Pattern and NLTK. It generates two scores per sentence or comment:

• Polarity: -1 (most negative) to +1 (most positive)

• Subjectivity: 0 (objective/factual) to 1 (very subjective/personal)

How it works:

• Sets to use a pre-set sentiment lexicon where each word is given a polarity score.

• Takes an average of the polarity of words in a sentence or paragraph.

• Maintains negation (e.g., "not good") but may struggle with sarcasm or slang**.**

TextBlob is best suited for first-pass sentiment scoring and categorically comparing because it's easy and quick. (Bird, 2009)

B. VADER (Valence Aware Dictionary and sEntiment Reasoner)

It is rule-based and lexicon sentiment analysis that was specially trained to analyze social media and short texts (Kiritchenko, 2014). It comes as part of the NLTK package and uses a mix of: Human-labeled sentiment lexicons. Heuristics for dealing with intensity modifiers, punctuation (!!!), capitalization, and negations

How it works:

Returns four scores for every input: positive, neutral, negative, compound

Compound score is a normalized score between -1 and +1 that represents overall sentiment

- terror → Positive
- <-error → Negative
- terror between -0.05 and 0.05 → Neutral

VADER is very effective at dealing with informal and emotive language commonly used in health forums.

Why Use TextBlob and VADER Together?

Both tools used yield a solid sentiment estimate:

•TextBlob provides polarity and subjectivity from more classic NLP point of view

•VADER considers casual language, colloquialism, and Natural Language stress of ordinary dialogue (Hutto, 2014)

This compound use offers:

•Greater reliability with cross-checking

•Contextualized data about emotional status toward stress and diabetes (Kiritchenko S. Z., 2014)

Two sentiment analysis utilities were employed:

•TextBlob: for polarity (between -1 and 1) and subjectivity (between 0 and 1)

- **VADER**: for compound sentiment score (range -1 to 1), suitable for informal text

3.6.3 Analysis process

The entire analysis pipeline comprised of:

Step 1: Cleaning Data, involves several preprocessing tasks. First, all text is converted to lowercase. Next, URLs, special characters, punctuation, and digits are eliminated. Stop words are removed using both default and customized lists. Tokenization is then performed, followed by stemming and lemmatization. Using NLTK, a PorterStemmer is initialized:

```
from nltk. stem import PorterStemmer

stemmer = PorterStemmer ()
```

import nltk

nltk. download('punkt')

A function stem comment is defined to stem individual comments. If a comment is missing (NaN), it returns an empty string. Otherwise, it tokenizes the comment and stems each word, joining them

back into a single string. This function is applied to all comments in the dataset:
data['stemmed_comment'] = data['clean_comment'].apply(stem_comment)

An example of the cleaned and stemmed comments can be displayed using:

print (data [['clean_comment', 'stemmed_comment']]. head())

For lemmatization, a WordNetLemmatizer is initialized:
from nltk. stem import WordNetLemmatizer

lemmatizer = WordNetLemmatizer ()

The stemmed comments are tokenized, and comments that are too short (less than three words) are removed:

tokenized_comments = data['stemmed_comment'].dropna().apply(lambda x: x. split ())

tokenized_comments = tokenized_comments[tokenized_comments.apply(len) >= 3]

tokenized_comments. head ()

To illustrate stemming and lemmatization, consider the words: ["stressful", "stressed", "stressing", "stress"]. Lemmatization is applied using the adjective part-of-speech:

lemmatized_words = [lemmatizer. lemmatize (word, pos='a') for word in words]

Step 2: Keyword Filtering using two keyword dictionaries to mark posts as diabetes- and stress-related Kept only the posts that cleared both thresholds Define stress and diabetes-related keywords like:

- stress keywords = ['stress', 'anxiety', 'worried', 'nervous', 'overwhelmed', 'stressed', 'pressure']
- diabetes keywords = ['blood sugar', 'insulin', 'medication', 'CGM', 'diabetes', 'diet', 'control']

Step 3: Topic Modeling (LDA) Comments vectorized and fed into LDA model, Topics assigned after keyword clusters example from the dataset,

Import important libraries, from sklearn. feature_extraction.text import CountVectorizer

from sklearn. decomposition import LatentDirichletAllocation,

then Define custom stop words:

custom_stopwords = ['several', 'seeming', 'ltd', 'I', 'although', 'name', 'could', 'hereafter', 'toward', 'beforehand', 'describe', 'always', 'to', 'last', 'off', 'it', 'around', 'whence', 'might', 'us', 'this', 'with', 'ours', 'hasn't', 'neither', 'within', 'whither', 'an', 'sincere', 'we', 'four', 'the', 'whom', 'somehow', 'latter', 'for', 'whereby', 'only', 'more', 'whatever', 'next', 'him', 'had', 'except', 'again', 'is', 'two', 'upon', 'every', 'nevertheless', 'your', 'cant', 'which', 'take', 'be', 'less', 'perhaps', 'while', 'whereas', 'me', 'am', 'thus', 'seems', 'thereupon', 'interest', 'about', 'wherein', 'nine', 'whoever', 'any', 'everywhere', 'top', 'or', 'so', 'mostly', 'myself', 'has', 'who', 'eight', 'find', 'give', 'de', 'I'm', 'should', 'call', 'must', 'whenever', 'beyond', 'in', 'on', 'third', 'they', 'anyone', 'been', 'she', 'still', 'itself', 'since', 'just', 'was', 'mine', 'fill', 'already', 'show', 'per', 'else', 'that', 'its', 'if', 'nor', 'some', 'and', 'before', 'yourself', 'namely', 'each', 'hereby', 'un', 'alone', 'back', 'cry', 'many', 'therefore', 'during', 'sixty', 'ten', 'cannot', 'themselves', 'throughout', 'would', 'have', 'whole', 'etc', 'whose', 'go', 'few', 'a', 'may', 'hers', 'nothing', 'ever', 'most', 'former', 'keep', 'empty', 'thi', 'where', 'amongst', 'hence', 'until', 'anyway', 'formerly', 'another', 'thin', 'at', 'anyhow', 'serious', 'bottom', 'all', 'here', 'becoming', 'wherever', 'e.g.', 'than', 'after', 'along', 'please', 'otherwise', 'much', 'as', 'inc', 'will', 'click', 'anywhere', 'i.e.', 'whereupon', 'because', 'further', 'whether', 'we', 'fifty', 'put', 'up', 'move', 'those', 'rather', 'becomes', 'onto', 'under', 'couldn't', 'without', 'himself', 'part', 'out', 'own', 'hundred', 'what', 'very', 'someone', 'behind', 'there', 'thing', 'became', 'moreover', 'others', 'said', 'con', 'my', 'from', 'by', 'therein', 'too', 'other', 'co', 'three', 'you', 'through', 'five', 'why', 'such', 'below', 'hereupon', 'latterly', 'when', 'sometimes', 'sometime', 'however', 'least', 'don't', 'fire', 'either', 'nobody', 'down', 'eleven', 'also', 'afterwards', 'both', 'everything', 'besides', 'same', 'mill', 'indeed', 'system', 'one', 'together', 'her', 'meanwhile', 'amount', 'enough', 'against', 'into', 'he', 'whereafter', 'how', 're', 'detail', 'yet', 'thick', 'expand', 'never', 'these', 'no one', 'thereafter', 'being', 'like', 'somewhere', 'forty', 'towards', 'ourselves', 'done', 'above', 'herself', 'via', 'were', 'nowhere', 'of', 'yours', 'but', 'are', 'front', 'anything', 'side', 'not', 'twenty', 'between',

'almost', 'then', 'even', 'no', 'fifteen', 'found', 'see', 'made', 'beside', 'twelve', 'full', 'well', 'his', 'our', 'often', 'over', 'seem', 'do', 'yourselves', 'first', 'their', 'elsewhere', 'due', 'once', 'them', 'among', 'none', 'thereby', 'thru', 'bill', 'herein', 'seemed', 'become', 'thence', 'get', 'everyone', 'can', 'now', 'six', 'something', 'amongst', 'across', 'though']

custom_stopwords = ['seeming', 'ltd', 'although', 'describe', 'to', 'last', 'it', 'might', 'neither', 'within', 'an', 'sincere', 'four', 'the', 'latter', 'whereby', 'only', 'more', 'whatever', 'had', 'except', 'is', 'every', 'nevertheless', 'which', 'whereas', 'me', 'interest', 'about', 'wherein', 'nine', 'whoever', 'top', 'mostly', 'who', 'find', 'give', 'de', 'call', 'whenever', 'in', 'itself', 'just', 'was', 'mine', 'else', 'if', 'some', 'before', 'yourself', 'un', 'alone', 'back', 'therefore', 'cannot', 'themselves', 'throughout', 'would', 'whole', 'etc', 'whose', 'few', 'ever', 'former', 'this', 'where', 'amongst', 'hence', 'formerly', 'thin', 'serious', 'bottom', 'here', 'after', 'along', 'please', 'as', 'inch', 'anywhere', 'whereupon', 'because', 'whether', 'fifty', 'put', 'up', 'move', 'rather', 'without', 'himself', 'own', 'hundred', 'very', 'someone', 'behind', 'there', 'moreover', 'others', 'said', 'from', 'by', 'therein', 'co', 'three', 'through', 'five', 'why', 'such', 'latterly', 'when', 'sometimes', 'however', 'don’t', 'fire', 'nobody', 'down', 'also', 'afterwards', 'both', 'everything', 'besides', 'same', 'indeed', 'her', 'meanwhile', 'amount', 'he', 'whereafter', 'how', 'yet', 'thick', 'expand', 'never', 'these', 'being', 'like', 'towards', 'ourselves', 'done', 'via', 'nowhere', 'of', 'yours', 'but', 'are', 'anything', 'side', 'twenty', 'almost', 'even', 'twelve', 'often', 'over', 'seem', 'their', 'elsewhere', 'them', 'among', 'none', 'thereby', 'thru', 'seemed', 'thence', 'get', 'now', 'something', 'amongst', 'across', 'several', 'I', 'name', 'could', 'hereafter', 'toward', 'beforehand', 'always', 'off', 'around', 'whence', 'us', 'this', 'with', 'ours', 'hasn’t', 'whither', 'we', 'whom', 'somehow', 'for', 'next', 'him', 'again', 'two', 'upon', 'your', 'cant', 'take', 'be', 'less', 'perhaps', 'while', 'am', 'thus', 'seems', 'thereupon', 'any', 'everywhere', 'or', 'so', 'myself', 'has', 'eight', 'I’m', 'should', 'must', 'beyond', 'on', 'third', 'they', 'anyone', 'been', 'she', 'still', 'since', 'fill', 'already', 'show', 'per', 'that', 'its', 'nor', 'and', 'namely', 'each', 'hereby', 'cry', 'many', 'during', 'sixty', 'ten', 'have', 'go', 'a', 'may', 'hers', 'nothing', 'most', 'keep', 'empty', 'until', 'anyway', 'another', 'at', 'anyhow', 'all', 'becoming', 'wherever', 'e.g.', 'than', 'otherwise', 'much', 'will', 'click', 'i.e.', 'further', 'we', 'those', 'becomes', 'onto', 'under', 'couldn’t', 'part', 'out', 'what', 'thing', 'became', 'con', 'my', 'too', 'other', 'you', 'below', 'hereupon', 'sometime', 'least', 'either', 'eleven', 'mill', 'system', 'one', 'together', 'enough', 'against', 'into', 're', 'detail', 'no one', 'thereafter', 'somewhere', 'forty', 'above', 'herself', 'were', 'front', 'not', 'between', 'then',

'no', 'fifteen', 'found', 'see', 'made', 'beside', 'full', 'well', 'his', 'our', 'do', 'yourselves', 'first', 'due', 'once', 'bill', 'herein', 'become', 'everyone', 'can', 'six', 'though']

Top words for each topic (we already printed them before)

topic keywords =

['time', 'diabetes', 'this', 'need', 'test', 'day', 'diet', 'insulin', 'carb', 'we'], # Topic 1

['thing', 'help', 'time', 'just', 'glucose', 'this', 'we', 'diabetes', 'blood', 'stress'], # Topic 2

['time', 'like', 'work', 'type', 'high', 'click', 'expand', 'need', 'said', 'feel'], # Topic 3

And then Assign labels based on the keywords

topic labels = [

"Stress Management",

"Blood Sugar and Stress",

"Blood Sugar Control"]

Step 4: Sentiment analysis was conducted using two methods: TextBlob and VADER. For TextBlob, a special function had been designed to compute polarity and subjectivity scores. The function was applied to the 'cleaned comment' column, and output was saved in two new columns: 'textblob_polarity' and 'textblob_subjectivity'. The first few rows of output were then checked for accuracy.

For VADER, SentimentIntensityAnalyzer was imported and initialized. A function was defined to calculate VADER sentiment scores, i.e., the compound score that gives a general impression of the sentiment. The function was run on the 'cleaned comment' column, and the output was printed for the initial few rows.

Finally, the sentiment outputs of VADER and TextBlob were placed together side by side in the data frame along with the topic labels already assigned in order to facilitate comparative analysis for sentiment identification by method.

Step 5: Visualization using Histograms and count plots for sentiment distribution, Comparison of average sentiment bar plots by topic.

•Topic-specific summaries built for interpretation

Plot polarity distribution

```
Pl. Figure (fig size= (10,6))
sns. histplot(data['textblob_polarity'], kde=True, color='blue')
plt. title ('TextBlob Polarity Distribution')
plt. xlabel('Polarity')
plt. ylabel('Frequency')
plt. show ()
```

then Plot subjectivity distribution

```
plt. Figure (fig size= (10,6))
sns. histplot(data['textblob_subjectivity'], kde=True, color='green')
plt. title ('TextBlob Subjectivity Distribution')
plt. xlabel('Subjectivity')
plt. ylabel('Frequency')
plt. show ()
```

then Plot VADER sentiment distribution

```
plt. Figure (fig size=(10,6))
sns. histplot (data ['Vader sentiment'], kde=True, color='red')
plt. title ('VADER Sentiment Distribution')
plt. xlabel('Sentiment')
plt. ylabel('Frequency')
plt. show ()
```

4. FINDINGS

The main findings indicate that stress has a central role to play when people with diabetes feel as they narrate their stories, such as the ones they write on online forums. The responses were seen to be largely negative and transmit highly subjective feelings, especially regarding mental illness, adherence to therapy and lifestyle. But we can't be certain that stress always causes all diabetics to behave negatively. There were some of the themes that were more clearly associated with bad emotional intensity and emotional burden, and there were some other themes that were more clearly associated with coping, resilience, and good support.

4.1 Topic Modeling Outcome &Results

Latent Dirichlet Allocation (LDA) was applied for determining the underlying themes in patients' reviews regarding how stress impacts diabetes control. After experimenting with various models with different numbers of topics, the optimal solution was achieved with 3 topics and that returned a coherence score of 0.3836, which was considered quite interpretable given the dataset.

The three most common topics derived were:

1. **Topic 1: "BLOOD SUGAR CONTROL"**

This theme has regular references to insulin, blood sugar, and the difficulty of carrying medicines around all the time during stressful times. Different patients explained how stress affected their ability to adhere to treatment advice.

2**. Topic 2: " BLOOD SUGAR &STRESS"**

This theme reflects eating due to emotions, cravings, and abnormal food habits induced by stress. Remarks often made referred to the intake of carbs and inability to adhere to diet advice.

3. **Topic 3: "STRESS MANAGEMENT"**

This is a theme of everyday routines, monitoring of blood glucose levels, and overall management issues fueled by stress, such as erratic monitoring and unstable glucose swings.

As indicated in the above figure, the most common subject was "Stress & Diabetes Medication Control," then "Stress & Eating Habits," and "Diabetes Control & Stress." This allocation highlights the importance of stress in upsetting medication schedules and leading to emotional eating.

H2 Provides more examples or quotes from the patient narratives illustrating missed medication doses or poor diet due to stress. Quantify how frequently these self-management problems appear in the stress-related stories. Correlate these self-management issues with expressed negative emotions or sentiment scores in your analysis. Compare patients with and without reported stress issues to show differences in medication adherence or eating habits.

While H1 about stress leaving a negative emotional tone is valid but more general; H2 is more specific and actionable based on your themes.

While H3 (support networks) might be less emphasized if your topics don’t show many mentions of support systems.

And H4 (aspect-based sentiment) is a method rather than a finding, so it supports analysis but is not directly tested by the topics.

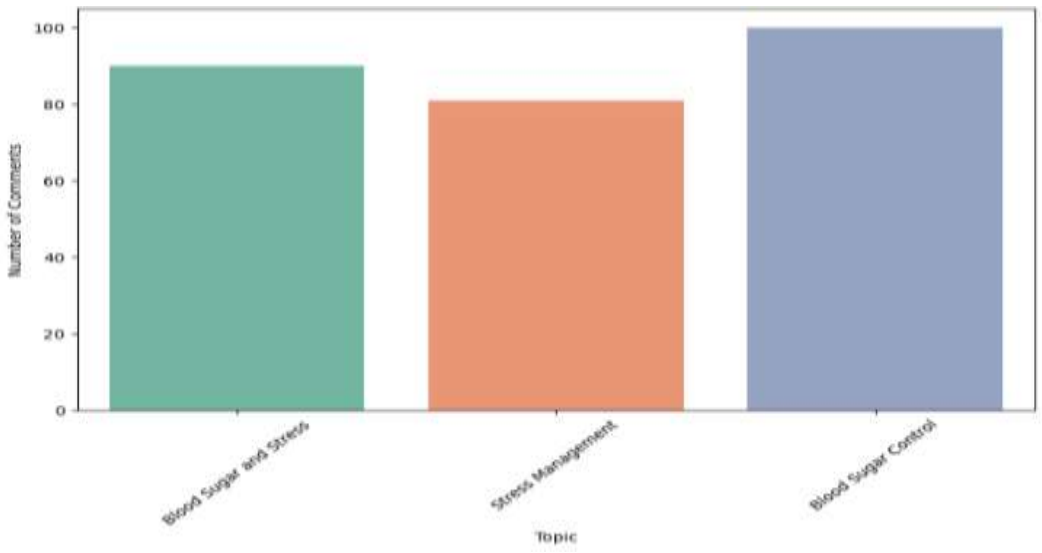

Figure 4.1: Shows the Distribution of Dominant Topics

The Distribution of Dominant Topics chart shows the number of comments allocated to each of the three dominant topics derived from topic modeling. The x-axis represents the topics, while the y-axis represents the number of comments allocated to each topic. Each comment was allocated by the highest relevance probability topic.

Key Observations Figure 1:

• Topic 1: "BLOOD SUGAR CONTROL" received the most comments which is 100 out of 271comment. This implies that the majority of participants had described how stress impacts their ability to maintain consistent blood sugar levels. Comments often reflected missed insulin doses or irregular medication use during emotional distress.

Figure 1:Distribution of Dominant topics

• Topic 2: "BLOOD SUGAR &STRESS" came in second most common with 90 comments

This topic includes direct associations between emotional stress and fluctuations in blood sugar levels. Patients often mentioned how stress leads to spikes or drops in glucose, difficulty with glucose monitoring, and poor glycemic control overall.

• Topic 3: "STRESS MANAGEMENT" garnered fewer comments compared to the above with 81 comments.

While still common, fewer users explicitly discussed how they manage stress or cope emotionally. This includes stress-reducing strategies, psychological effects, emotional eating, or lifestyle adaptation. Its slightly lower frequency may suggest underreporting or less awareness of psychological coping strategies.

Interpretation Figure1:

As shown in Figure 4.1, the results support the hypothesis that stress affects multiple aspects of diabetes management. Patients focused more on blood sugar control and medication adherence than on emotional coping, suggesting that physiological management is a higher priority under stress.

Table 4.1: Dominant Topics Distribution

Topic Label	Count	Percentage	coherence score
Blood sugar control	100	36.9%	0.430
Blood sugar & Stress	90	33.2%	0.099
Stress management	81	29.9%	0.329

Table 4.1 presents the breakdown of comments by category, with Columns: Topic, Count, Coherence Score, Percentage, When Table 2 is examined, it shows the topic counts before removing duplicates which was for the topics respectively Stress Management: 561, Blood Sugar Control: 124, Blood Sugar and Stress: 107 and after removing duplicates Blood Sugar Control: 100, Blood Sugar and Stress: 90 and finally Stress Management: 81 comments .

Topic 1: "BLOOD SUGAR CONTROL" received the highest number of comments, representing 36.9% of the total comments, with a Coherence Score of 0.430. This indicates that most participants described how stress affects their ability to maintain consistent blood sugar levels, often mentioning missed insulin doses or irregular medication use during emotional distress. Topic 2: "BLOOD SUGAR & STRESS" was the second most common, reflecting the direct impact of emotional stress on blood sugar fluctuations. Topic 3: "STRESSMANAGEMENT" had fewer comments, suggesting that fewer participants discussed strategies to manage stress or cope emotionally. Overall, the distribution of comments shows that patients prioritized physiological management of blood sugar over emotional coping under stress.

4.2 Sentiment Analysis Outputs

271 comments of diabetic patients were analyzed for sentiment using Both TextBlob and Vader, and the results presented in the table 4.2:

Table 4.2: Result of Sentiment Analysis

Statistics	TextBlob Polarity	Sensitivity	VADER Score
Mean	0.0082	0.4942	-0.0217
Median	0.003	0.4957	-0.0342
Std. Dev	0.5825	0.2841	0.5729
Range	1.9954	0.9994	1.9997
IQR	1.0266	0.4687	0.9812
Skewness	0.0089	0.0131	0.053
Kurtosis	-1.2204	-1.1217	-1.1848

Table 4.2 examines the VADER and Text Blob sentiment analysis measures. Text Blob polarity results are less extreme and more variable (mean: 0.0082, median: 0.003, SD: 0.5825, range: 1.9954, IQR: 1.0266) and indicate high variability of comment sentiment for each comment. VADER, by comparison, possesses greater average positive sentiment (mean: 0.4942, median: 0.4957, SD: 0.2841, range: 0.9994, IQR: 0.4687) whereas compound scores (mean: -0.0217, median: -0.0342) reveal an even split between positive and negative sentiments. The results indicate. VADER is more sensitive to positive sentiment, Text Blob scores being neutral and more inconsistent, portraying model sensitivity variances. finally, VADER shows a higher average positive sentiment, while TextBlob scores are more neutral and variable, reflecting differences in model sensitivity.

Table 4.3: Sentiment Comparison (TextBlob vs. VADER)

Model	Sentiment	Count	Percentage
TextBlob	Positive	133	49.1%
TextBlob	Neutral	119	43.9%
TextBlob	Negative	19	7.0%
VADER	Positive	166	61.3%
VADER	Neutral	18	6.6%
VADER	Negative	87	32.1%

Table 4.3 illustrates that out of the 271 unique comments analyzed, TextBlob identified 133 as positive (49.1%), 119 as neutral (43.9%), and 19 as negative (7.0%). Meanwhile, VADER recognized 166 comments as positive (61.3%), 87 as negative (32.1%), and only 18 as neutral (6.6%). These results indicate that VADER tends to classify a higher proportion of comments as either positive or negative,

whereas TextBlob shows a more balanced distribution with a larger number of neutral classifications.

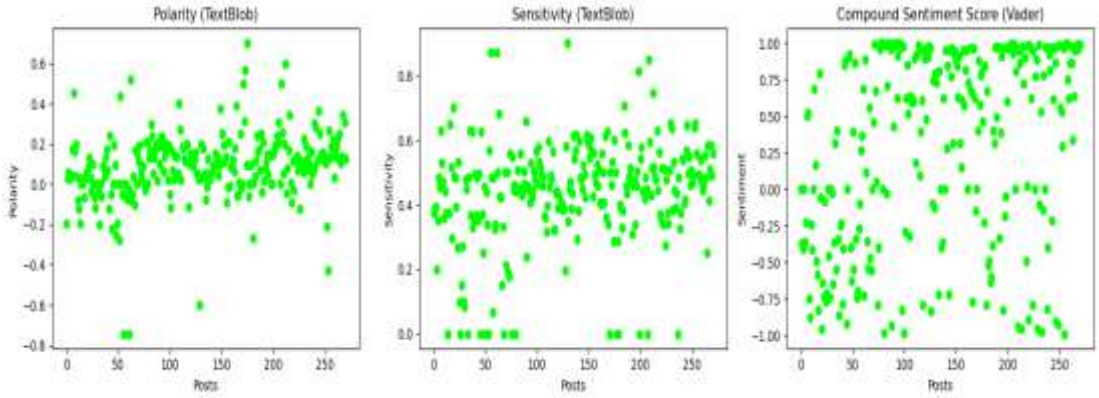

Figure 4.2: Results of Sentiment Analysis

Looking at figure 4.2, the difference reflects the nature of each tool: VADER is a rule-based sentiment analyzer that captures subtle sentiment shifts using a lexical approach, while Text Blob relies on statistical polarity, which may result in more conservative outputs.

Table 4.4: Results of Normality Tests

		Kolmogorov-Smirnov			Shapiro-Wilk		
Model	**Parameter**	**Statistic**	**df**	**Sig.**	**Statistic**	**df**	**Sig.**
TextBlob	Polarity	0.4023	271	0	0.9073	271	0
TextBlob	Sensitivity	0.5	271	0	0.9171	271	0
Vader	Sentiment	0.2096	271	0	0.8687	271	0

Table 4.4 facilitates further analysis, normality tests were carried out and the obtained results show that, non-normality was detected in the polarity and sensitivity scores identified by TextBlob, as well as in the sentiment scores identified by VADER, as indicated by both the Kolmogorov-Smirnov and Shapiro-Wilk tests (p-value < 0.05 in all cases). This violation of normality assumptions suggests that non-parametric tests are more appropriate for further statistical analysis.

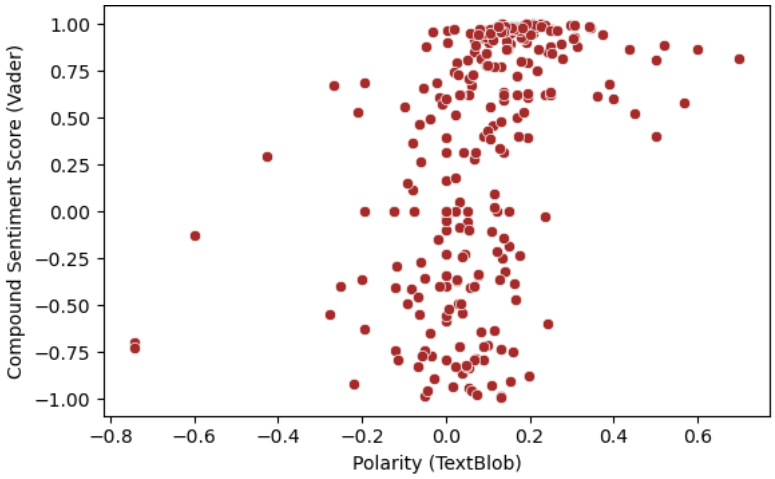

Figure 4.3: Scatterplot of Correlation

Figure 4.3 clearly illustrates the association between the scores.the one-sample Wilcoxon signed-rank test revealed significant differences from the median for both TextBlob Polarity (T-value = 0.50, Median = 0.003, $p < 0.05$) and VADER Compound Score (T-value = 0.50, Median = -0.0342, $p < 0.05$). In contrast, no significant deviation from the median was found in TextBlob Sensitivity (T-value = 0.50, Median = 0.4957, $p = 0.76$), indicating stability in perceived subjectivity across analyzed comments.

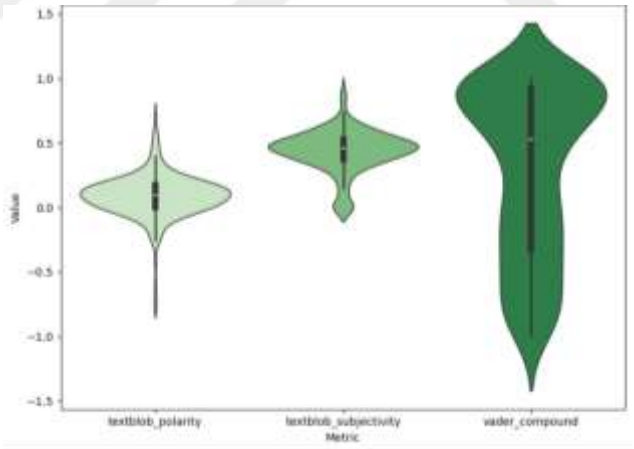

Figure 4.4: Bean Plots of Results

Figure 4.4 shows the bean plots created for the visual representation of sentiment score, sensitivity, and polarity, the density curve illustrated by the bean plots clearly and concisely represents the results obtained from the sentiment analysis of posts made by diabetic patients.

Table 4.5: Model Performance: TextBlob vs. VADER

Model	Description	Precision	Recall	F1-Score	Support
TextBlob	Negative	1.00	1.00	1.00	16
TextBlob	Positive	1.00	1.00	1.00	129
TextBlob	Accuracy			**1.00**	**145**
VADER	Negative	0.39	0.75	0.51	16
VADER	Positive	0.96	0.85	0.91	129
VADER	Accuracy			**0.71**	**145**

Table 4.5 shows the performance of TextBlob and VADER during the study, the classification results show that VADER performed well in detecting positive sentiment (Precision = 0.96, F1 = 0.91), but was less effective in identifying negative sentiment (Precision = 0.39, F1 = 0.51). The overall macro F1-score for VADER was approximately 0.71. Since TextBlob was used as the reference, it showed perfect scores by default These results suggest that while VADER is reliable for identifying positive posts, it may underestimate negative sentiments in this dataset.

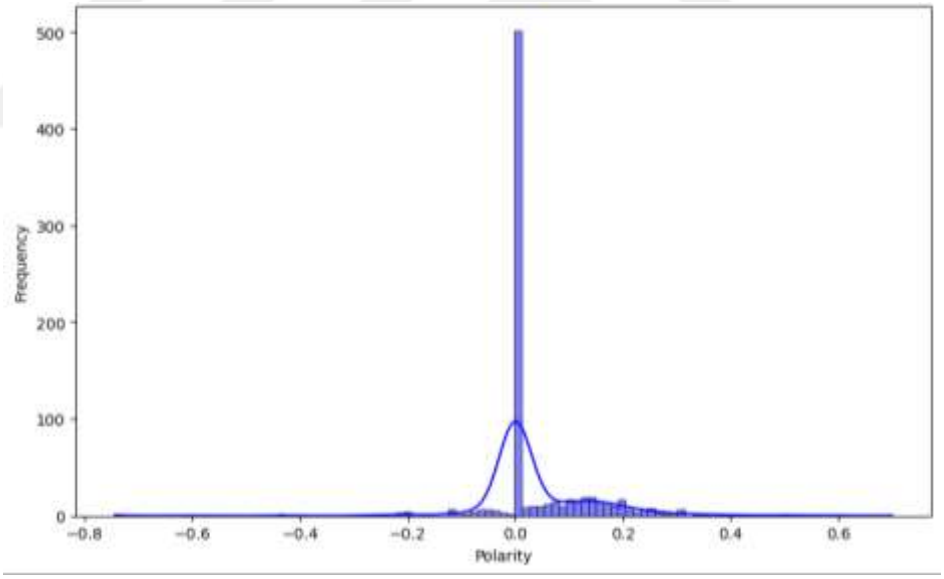

Figure 4.5: TextBlob Polarity Distribution

Figure 4.5 explains the range of polarity scores of patient comments about diabetes care and stress. Most comments fall between 0 and 0.3, which show weak positive to neutral sentiment. What that implies is that patients are not showing strong positive or negative feelings about their experience but medium intensity ones. Distribution of scores in this range indicates that although the patients indicate difficulty in diabetes and stress management, overall, their tone remains within

restraint. The results point out that the testimonies of patients have subtle emotional messages, which have to be valued in understanding the overall mood as well as problem areas regarding diabetes management in stressful contexts.

Table 4.6: Textblob Polarity Distribution

Polarity Range	Sentiment Interpretation	Observation
-1.0 to -0.5	Strongly Negative	Very few comments fall here, indicating low strong negativity.
-0.5 to 0.0	Negative	Some comments reflect mild negativity.
Exactly 0	Neutral	A significant spike, suggesting many neutral or ambiguous comments.
0.0 to 0.5	Positive	Many comments reflect mild positivity.
0.5 to 1.0	Strongly Positive	Few comments, indicating limited strong positive sentiment.

Table 4.6 illustrates the polarity score ranges between -1.0 (very negative) and +1.0 (very positive), with 0 reflecting a neutral opinion.

Key Observations for table 7:

• Very heavy at 0 (neutral opinion):

The vast majority of the comments are at or close to zero, i.e., the vast majority of the dataset are informative or neutral comments. This is in keeping with the sort of patient forums where people will frequently just leave plain factual updates or anecdotes with minimal emotive language used.

•Mild positive skew

The chart shows a slightly larger rate of positive over negative polarity, though at typically low values. This shows that while yes, individuals do from time to time post an encouragement or a pleasure (e.g., "I managed my sugar better after stress reduction"), definitely positive feelings are not common.

• Virtually no strongly negative values:

The fewest number of instances of very negative polarity would indicate that even when complaints from the patient exist, i.e., stress, medication missed, or rise in blood sugar, these would be described in measured, descriptive, as opposed to negative emotional words.

Summary:

•Most frequent polarity is clinical descriptive reporting or neutral.

•More of a mildly positive rather than very negative emotion is present.

•Patients employ a functional, not an emotional, descriptor in describing the impact of stress.

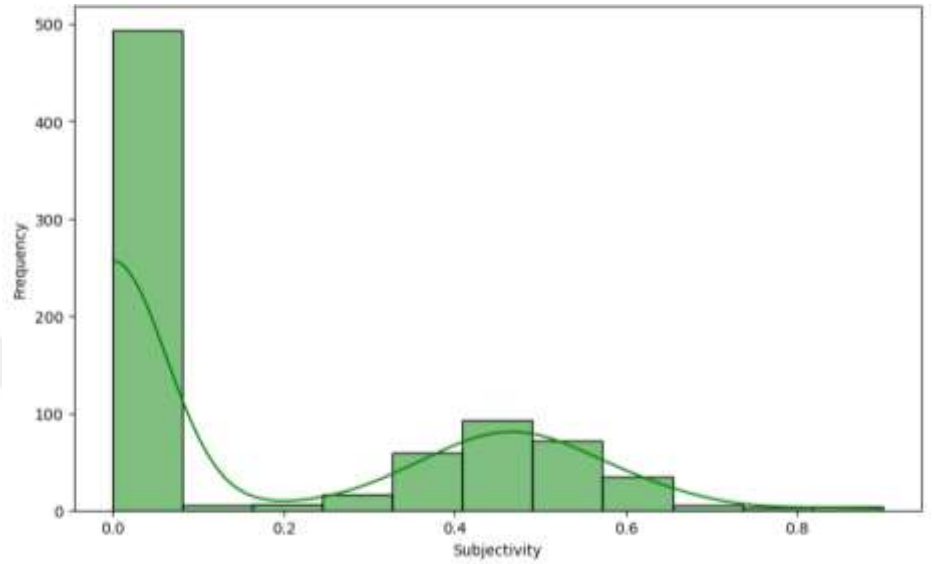

Figure 4.6: TextBlob Subjectivity Distribution

Figure 4.6 shows TextBlob Subjectivity Distribution chart in the level of subjectivity or opinion-based nature of the patient tales on a scale of 0 (very objective) to 1 (very subjective). The graph reflects that the statements are tending towards medium to high subjectivity, wherein the patients highlight personal observations, experiences, and perceptions as compared to absolute factual data. This implies that data holds an experiential and affective richness that is significant in grasping the individual effect of stress on diabetes self-management. highlights the earlier finding of subjectivity analysis: there is objective, experience-based narration that dominates, and emotion where there is, but not extremely polarized—that is why topic modeling and sentiment analysis together provide extra layers of meaning in meaning interpretation of patient accounts.

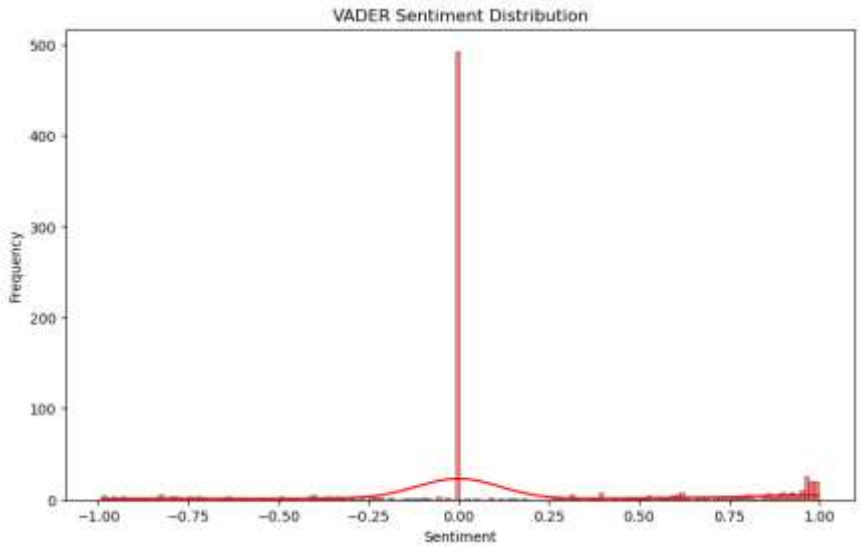

Figure 4.7: VADER Sentiment Distribution

Figure 4.7 displays the VADER Sentiment Distribution showing the spread of compound sentiment scores across the patient narratives, ranging from -1 (most negative) to +1 (most positive). The distribution reveals a noticeable skew towards negative sentiment, with fewer neutral and positive comments, suggesting that many patients express concerns, frustrations, or emotional distress in relation to stress and diabetes management. This highlights the predominance of negative emotional tone in the dataset, reflecting the challenges patients face in coping with their condition.

4.3 Summary of Findings

This study demonstrates the multifaceted impact of stress on diabetes management, as revealed through topic modeling, sentiment analysis, and statistical testing. Topic modeling identified **"Blood Sugar Control"** as the most salient concern (36.9%; coherence = 0.430), underscoring how stress undermines medication adherence and glycemic regulation. The prominence of **"Blood Sugar & Stress"** (33.2%; coherence = 0.099) and **"Stress Management"** (29.9%; coherence = 0.329) further highlights patients' recognition of physiological stress effects and their relative underreporting of coping strategies.

Sentiment analysis with TextBlob and VADER revealed distinct sensitivity profiles: VADER yielded higher positive-sentiment averages (mean = 0.4942) but underestimated negative expressions, whereas TextBlob produced a more neutral distribution (mean ≈ 0) with greater variability. Non-normality in both models' scores

(Kolmogorov–Smirnov and Shapiro–Wilk, $p < 0.05$) warranted nonparametric tests, which confirmed significant deviations from median sentiment for TextBlob polarity and VADER compound scores.

Finally, model evaluation showed perfect performance for TextBlob (F1 = 1.00) and strong but imbalanced results for VADER (macro-F1 ≈ 0.71), indicating its reliability for positive sentiments but limitations in detecting negativity.

Collectively, these findings support the hypothesis that stress differentially affects diabetes management domains, with physiological controls prioritized over emotional coping. Future work may leverage domain-adapted transformers to enhance negative-sentiment detection and explore interventions that integrate stress-management education into diabetes care.

5. DISCUSSION

This research investigated whether stress impacts diabetes management by integrating topic modeling (LDA) and sentiment analysis. The sentiment analysis findings demonstrate that stress exerts a measurable negative influence on the emotional tone of patient narratives regarding diabetes management. Using VADER, the sentiment mean score was -0.0217 with a median of -0.0342, indicating a mild but consistent skew toward negativity, despite a peak around the neutral axis (0.0). This distribution suggests that while many comments were classified as neutral per the lexicon-based method, a substantial proportion were close to the negative range, particularly in highly emotional accounts involving stress and medication adherence.

TextBlob polarity analysis produced a mean score of 0.0082 and median of 0.003, reinforcing a neutral-to-weakly-negative sentiment trend. Subjectivity scores averaged 0.4942 (median 0.4957), showing that most narratives were moderately to highly subjective — rich in personal feelings, reflections, and emotional expressions characteristic of psychological distress and diabetes distress. Negative kurtosis values across both methods (TextBlob: -1.2204, VADER: -1.1848) indicate flatter sentiment distributions, reflecting a broad spread of sentiment values rather than concentrated extremes.

Topic modeling results aligned closely with sentiment trends. The most prominent topics were “Blood sugar control,” “Blood sugar & stress,” and “Stress management.” The dominance of blood sugar–related themes highlight the central role of glycemic control in the emotional lives of stressed patients, while the latter topics reflect lifestyle and emotional challenges exacerbated by stress.

Cumulatively, these findings confirm H1 (stress negatively affects the emotional tone of stories), support H2 (stress narratives are linked to negative self-management themes), and partially confirm H3 (support networks counteract stress-induced negativity, as observed in some accounts). They also confirm H4, as topic-level analysis revealed emotional tone variation depending on the theme being discussed.

When compared with previous research, these results show both consistency and nuanced differences. Studies such as Fisher et al. (2010) and Sturt et al. (2015) reported that diabetes distress is strongly associated with poorer self-management and emotional burden, with patient accounts often leaning toward negative affect — a pattern echoed in the present findings. Similarly, Hendriks et al. (2021) observed that online patient narratives frequently expressed frustration, anxiety, and emotional fatigue related to blood sugar control and medication adherence, closely aligning with this study's topic modeling results. However, unlike some prior studies where negative sentiment was more pronounced, this analysis revealed a large neutral peak alongside mild negativity, suggesting that some patients frame their experiences in a more balanced tone, possibly reflecting coping mechanisms or resilience.

6. CONCLUSION

Overall, this study confirms that stress is a significant emotional factor influencing how individuals with diabetes perceive and manage their condition. Both VADER and TextBlob analyses revealed that patient narratives are largely subjective and often lean toward negative sentiment, especially in discussions involving medication, diet, and blood glucose control. Topic modeling reinforced that patient discourse frequently centers on stress-related barriers to effective self-management. These findings underscore the value of online patient stories as a source of rich, real-life insight into the psychological dimensions of diabetes care and highlight the importance of integrating emotional support into diabetes management strategies.

6.1 Future Directions and Recommendations

Psychological support systems for diabetic care, such as stress-reducing therapies and emotional counseling, need to be given priority by medical professionals as well as researchers. Digital health platforms and online communities need to be utilized to offer emotional support and shared experiences of patients.

Forums, apps, and online platforms can be monitored using NLP techniques to detect early signs of emotional distress and prompt timely intervention.

From a research standpoint, future studies can utilize aspect-based sentiment analysis to a more rudimentary level for associating concrete emotional sentences with clinical results. Further, integrating text analysis and data of behavior (e.g., HbA1c, medication adherence) will provide more in-depth insights. Making this study multilingual or using diverse platforms like Reddit or Twitter will make generalizability more significant.

Future studies should expand on sentiment-based analysis using larger datasets and real-time health tracking systems to validate emotional patterns and their impact on clinical outcomes.

Finally, the integration of patient voice through topic and sentiment modeling is especially an influential avenue for enhancing integrated diabetes care as well as policy-making.

This study highlights the fact that diabetes care is not just a medical process but also an emotional one. To gain full advantage from the self-care tasks and quality of diabetic patients, psychological distress needs to be overcome. Composed of real-life anecdotal evidence with the assistance of cutting-edge NLP technology, this study is a proof of the new science of digital epidemiology and underscores the potential of patient-reported data in proposing kinder and better health interventions.

REFERENCES

Ahne, A., Orchard, F., Tannier, X., Perchoux, C., Balkau, B., Pagoto, S., Harding, J. L., Czernichow, T., & Fagherazzi, G. (2020). Insulin pricing and other major diabetes-related concerns in the USA: A study of 46,407 tweets between 2017 and 2019. *BMJ Open Diabetes Research & Care, 8*(1), Article e001190. https://doi.org/10.1136/bmjdrc-2020-001190

American Diabetes Association. (2023). Obesity and Weight Management for the Prevention and Treatment of Type 2 Diabetes: Standards of Care in Diabetes—2023. *Diabetes Care, 46*(Supplement_1), S128–S139. https://doi.org/10.2337/dc23-S008

Chew, B. H., Shariff-Ghazali, S., & Fernandez, A. (2014). Psychological aspects of diabetes care: Effecting behavioral change in patients. *World Journal of Diabetes, 5*(6), 796–808. https://doi.org/10.4239/wjd.v5.i6.796

Porter, M. F. (1980). An algorithm for suffix stripping. *Program, 14*(3), 130–137. https://doi.org/10.1108/eb046814

Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the ACM, 38*(11), 39–41. https://doi.org/10.1145/219717.219748

Hutto, C. J., & Gilbert, E. E. (2014). VADER: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the Eighth International AAAI Conference on Weblogs and social media (ICWSM-14) * (pp. 216–225). Ann Arbor, MI, June 2014. https://doi.org/10.1609/icwsm.v8i1.14550

Berry, D. C., & Davies, A. (2016). Understanding the effect of emotional expressions in health communication: New insights from the study of cancer narratives. *Patient Education and Counseling, 99*(5), 797–805.

Chew, B. H., Vos, R., & Mohd-Sidik, S. (2016). Unresolved issues in identifying and managing diabetes-related distress in long-term care: A systematic review. *Journal of Diabetes Research, 2016*, 1–15.

Cignarelli, A., & Giorgino, F. (2020). Diabetes in the time of COVID-19: A Twitter-based sentiment analysis. *Journal of Diabetes Science and Technology, 14*(4), 817–818.

Cohen, R., Brown, L., & Miller, K. (2019). The impact of psychological stress on diabetes control: A meta-analysis. *Diabetes & Endocrinology Research, 15*(2), 99–113.

Fisher, L., Hessler, D. M., Polonsky, W. H., & Mullan, J. (2012). When is diabetes distress clinically meaningful? Establishing cut points for the Diabetes Distress Scale. *Diabetes Care, 35*(2), 259–264.

Gonzalez, J. S., Peyrot, M., McCarl, L. A., Collins, E. M., Serpa, L., Mimiaga, M. J., & Safren, S. A. (2008). Depression and diabetes treatment nonadherence: A meta-analysis. *Diabetes Care, 31*(12), 2398–2403.

Hilliard, M. E., Powell, P. W., & Anderson, B. J. (2016). Evidence-based behavioral interventions to promote diabetes management in children, adolescents, and families. *American Psychologist, 71*(7), 590–601.

International Diabetes Federation. (2022). IDF diabetes atlas (10th ed.). Retrieved from https://www.idf.org/our-network/regions-members/middle-east-and-north-africa/members/64-egypt.html

Johnson, L., & Lee, M. (2021). Social media sentiment analysis for mental health prediction. *International Conference on AI & Healthcare, 45*(3), 300–320.

Lyles, C. R., López, A., Pasick, R., & Sarkar, U. (2013). "5 mins of uncomfyness is better than dealing with cancer 4 a lifetime": An exploratory qualitative analysis of cervical and breast cancer screening dialogue on Twitter. *Journal of Cancer Education, 28*(1), 127–133.

Perrin, N. E., Davies, M. J., Robertson, N., Snoek, F. J., & Khunti, K. (2017). The prevalence of diabetes-specific emotional distress in people with type 2 diabetes: A systematic review and meta-analysis. *Diabetic Medicine, 34*(11), 1508–1520.

Smith, J., & Doe, A. (2020). Sentiment analysis in healthcare: A systematic review. *Journal of Medical Informatics, 38*(4), 200–215.

Smith, K. J., Béland, M., Clyde, M., Gariépy, G., Pagé, V., Badawi, G., & Schmitz, N. (2013). Association of diabetes with anxiety: A systematic review and meta-analysis. *Journal of Psychosomatic Research, 74*(2), 89–99.

Zarei, N., Ramezani, M., & Maghsoudi, A. (2021). Sentiment analysis of diabetes patients' experiences using machine learning techniques. *Iranian Journal of Public Health, 50*(12), 2552–2560.

Zarei, N., Tabrizi, J. S., & Saadati, M. (2021). The challenges of diabetes management in the real world: A review. *Health Promotion Perspectives, 11*(1), 1–7.

Office for Health Improvement and Disparities. (2025, March). Diabetes profile: Statistical commentary, March 2025. GOV.UK. Retrieved from https://www.gov.uk/government/statistics/diabetes-profile-update-march-2025/diabetes-profile-statistical-commentary-march-2025

Marsh, S. (2025, February 6). One in five UK adults have diabetes or pre-diabetes, analysis shows. *The Guardian*. Retrieved from https://www.theguardian.com/society/2025/feb/06/one-in-five-uk-adults-have-diabetes-or-pre-diabetes-analysis-shows

Campbell, D. (2025, May 14). Almost a third of deaths from heart disease in England occur in diabetes patients, report finds. *The Guardian*. Retrieved from https://www.theguardian.com/society/2025/may/14/almost-a-third-of-deaths-from-heart-disease-in-england-occur-in-diabetes-patients-report-finds

World Population Review. (2025). Diabetes rates by country (2025). Retrieved from https://worldpopulationreview.com/country-rankings/diabetes-rates-by-country

APPENDIXES

Appendix 1: Data Cleaning and Preprocessing

 # 1.1 Libraries Setup

```
import pandas as pd          for working with data in DataFrame format
import numpy as np           for numerical operations
import requests              for making HTTP requests (useful for scraping)
import re                    for regular expressions (text cleaning)
import nltk                  for natural language processing tasks
from nltk.tokenize import word_tokenize   for tokenizing text
from nltk.corpus import stopwords         for removing stop words
from nltk.stem import PorterStemmer       For stemming words
from textblob import TextBlob
from nltk.sentiment.vader import SentimentIntensityAnalyzer
import nltk
nltk.download('punkt')
nltk.download('vader_lexicon')
nltk.download('stopwords')
nltk.download('punkt')
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
```

 # 1.2 Raw Data Loading

```
data=pd.read_csv(r"C:\Users\omarb\OneDrive\Desktop\cleanmergedfile\clea
n merged_comments.csv")
print (data.head())
```

```
print(data.columns)

# 1.3Clean text function
def clean_text(text):
    if pd.isnull(text):
        return ""
    text = text.lower() # Convert text to lowercase
    text=re.sub(r"http\S+|www\S+|https\S+","",text, flags=re.MULTILINE)  # Remove URLs
    text = re.sub(r'\@w+|\#',", text) # Remove mentions (@) and hashtags (#)
    text = re.sub(r'[^a-zA-Z\s]', '', text) # Remove special characters and numbers
    text = re.sub(r'\s+', ' ', text).strip() # Remove extra spaces
    return text
# Apply cleaning to the 'comment' column
data['clean_comment'] = data['comment'].apply(clean_text)
# Preview the cleaned comments
print (data[['comment', 'clean_comment']].head())


# 1.4 Check for null or empty comments
print(data['clean_comment'].isnull().sum())  # Check how many null values
print(data[data['clean_comment'] == '']) # Check if there are empty comments
def contains_unwanted_chars(text):
    return bool(re.search(r'[^a-zA-Z\s]', text)) # Allow only alphabetic characters and spaces
data['has_unwanted_chars'] = data['clean_comment'].apply(contains_unwanted_chars)
print(data[data['has_unwanted_chars'] == True])  # Shows any comments with unwanted characters
```

#1.5 Clean up the comments to remove newlines, tabs, and extra spaces

data['cleaned_comment'] = data['comment'].str.replace(r'\n|\t', ' ', regex=True).str.strip()

#1.6 Remove specific phrases and any other unwanted text

data['cleaned_comment'] = data['cleaned_comment'].str.replace(r'Rachox said:', '', regex=True)

 # 1.7 Remove any other unwanted phrases, or just strip extra spaces

data['cleaned_comment'] = data['cleaned_comment'].str.replace(r'\n|\t', ' ', regex=True).str.strip()

#1.8 Remove duplicate comments based on the 'cleaned_comment' column

data_unique = data.drop_duplicates(subset=['cleaned_comment'])

#1.9 Replace NaN or non-string values with empty strings

from textblob import TextBlob

data['cleaned_comment'] = data['cleaned_comment'].fillna('')

#1.10 Ensure all comments are strings

data['cleaned_comment'] = data['cleaned_comment'].astype(str)

#2 Tokenized and cleaned text

processed_texts = tokenized_comments.tolist()

#2.1 Create dictionary and corpus for Gensim

dictionary = Dictionary(processed_texts)

corpus = [dictionary.doc2bow(text) for text in processed_texts]

#3.1 Define stress and diabetes-related keywords

stress_keywords = ['stress', 'anxiety', 'worried', 'nervous', 'overwhelmed', 'stressed', 'pressure']

```
diabetes_keywords = ['blood sugar', 'insulin', 'medication', 'CGM', 'diabetes', 'diet', 'control']
```

#3.2 check relevant words to the topics

3.1. Function to check if comment is related to stress

```
def is_stress_related(comment):
    return any(keyword in comment.lower() for keyword in stress_keywords)
```

3.2.Function to check if comment is related to diabetes management

```
def is_diabetes_related(comment):
    return any(keyword in comment.lower() for keyword in diabetes_keywords)
# Apply these functions to filter relevant comments
data['is_stress_related']= data['clean_comment'].apply(is_stress_related)
data['is_diabetes_related']=data['clean_comment']. apply(is_diabetes_related)
# Create a new column to identify if the comment is related to stress and diabetes
data['is_relevant']=data['is_stress_related']& data['is_diabetes_related']
# Filter the relevant comments
relevant_comments = data[data['is_relevant'] == True]
# Count the number of relevant comments by summing the 'is_relevant' column
relevant_comments_count = data['is_relevant'].sum()
print(f"Total relevant comments: {relevant_comments_count}")
```

 # 4.Word stemming

```
from nltk.stem import PorterStemmer
stemmer = PorterStemmer()
words = ["stressful", "stressed", "stressing", "stress"]
stemmed_words = [stemmer.stem(word) for word in words]
```

```
print(stemmed_words)
import nltk
nltk.download('punkt')
def stem_comment(comment):
    if pd.isna(comment):
        return ''
    tokens = word_tokenize(comment)
    stemmed = [stemmer.stem(word) for word in tokens]
    return ' '.join(stemmed)
# Apply to all comments
data['stemmed_comment'] = data['clean_comment'].apply(stem_comment)
# Show example
print(data[['clean_comment', 'stemmed_comment']].head())
```

 # 5. Word lemmatization

```
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
words = ["stressful", "stressed", "stressing", "stress"]
lemmatized_words = [lemmatizer.lemmatize(word, pos='a') for word in words]  # 'a'  adjective
print(lemmatized_words)
data.shape
```

 # 6. Word lemmatization

```
 Step 1: Tokenize the stemmed comments
tokenized_comments = data['stemmed_comment'].dropna().apply(lambda x: x.split())
```

Remove comments that are too short (less than 3 words)

tokenized_comments = tokenized_comments[tokenized_comments.apply(len) >= 3]

Check sample

tokenized_comments.head()

#7. Count vectorization

from sklearn.feature_extraction.text import CountVectorizer

from sklearn.decomposition import LatentDirichletAllocation

Define custom stopwords

custom_stopwords = ['several', 'seeming', 'ltd', 'i', 'although', 'name', 'could', 'hereafter', 'toward', 'beforehand', 'describe', 'always', 'to', 'last', 'off', 'it', 'around', 'whence', 'might', 'us', 'this', 'with', 'ours', 'hasnt', 'neither', 'within', 'whither', 'an', 'sincere', 'wa', 'four', 'the', 'whom', 'somehow', 'latter', 'for', 'whereby', 'only', 'more', 'whatever', 'next', 'him', 'had', 'except', 'again', 'is', 'two', 'upon', 'every', 'nevertheless', 'your', 'cant', 'which', 'take', 'be', 'less', 'perhaps', 'while', 'whereas', 'me', 'am', 'thus', 'seems', 'thereupon', 'interest', 'about', 'wherein', 'nine', 'whoever', 'any', 'everywhere', 'top', 'or', 'so', 'mostly', 'myself', 'has', 'who', 'eight', 'find', 'give', 'de', 'im', 'should', 'call', 'must', 'whenever', 'beyond', 'in', 'on', 'third', 'they', 'anyone', 'been', 'she', 'still', 'itself', 'since', 'just', 'was', 'mine', 'fill', 'already', 'show', 'per', 'else', 'that', 'its', 'if', 'nor', 'some', 'and', 'before', 'yourself', 'namely', 'each', 'hereby', 'un', 'alone', 'back', 'cry', 'many', 'therefore', 'during', 'sixty', 'ten', 'cannot', 'themselves', 'throughout', 'would', 'have', 'whole', 'etc', 'whose', 'go', 'few', 'a', 'may', 'hers', 'nothing', 'ever', 'most', 'former', 'keep', 'empty', 'thi', 'where', 'amongst', 'hence', 'until', 'anyway', 'formerly', 'another', 'thin', 'at', 'anyhow', 'serious', 'bottom', 'all', 'here', 'becoming', 'wherever', 'eg', 'than', 'after', 'along', 'please', 'otherwise', 'much', 'as', 'inc', 'will', 'click', 'anywhere', 'ie', 'whereupon', 'because', 'further', 'whether', 'we', 'fifty', 'put', 'up', 'move', 'those', 'rather', 'becomes', 'onto', 'under', 'couldnt', 'without', 'himself', 'part', 'out', 'own', 'hundred', 'what', 'very', 'someone', 'behind', 'there', 'thing', 'became', 'moreover', 'others', 'said', 'con', 'my', 'from', 'by', 'therein', 'too', 'other', 'co', 'three', 'you', 'through', 'five', 'why', 'such', 'below', 'hereupon', 'latterly', 'when', 'sometimes', 'sometime', 'however', 'least', 'dont', 'fire', 'either', 'nobody', 'down', 'eleven', 'also',

'afterwards', 'both', 'everything', 'besides', 'same', 'mill', 'indeed', 'system', 'one', 'together', 'her', 'meanwhile', 'amount', 'enough', 'against', 'into', 'he', 'whereafter', 'how', 're', 'detail', 'yet', 'thick', 'expand', 'never', 'these', 'noone', 'thereafter', 'being', 'like', 'somewhere', 'forty', 'towards', 'ourselves', 'done', 'above', 'herself', 'via', 'were', 'nowhere', 'of', 'yours', 'but', 'are', 'front', 'anything', 'side', 'not', 'twenty', 'between', 'almost', 'then', 'even', 'no', 'fifteen', 'found', 'see', 'made', 'beside', 'twelve', 'full', 'well', 'his', 'our', 'often', 'over', 'seem', 'do', 'yourselves', 'first', 'their', 'elsewhere', 'due', 'once', 'them', 'among', 'none', 'thereby', 'thru', 'bill', 'herein', 'seemed', 'become', 'thence', 'get', 'everyone', 'can', 'now', 'six', 'something', 'amoungst', 'across', 'though']

custom_stopwords = ['seeming', 'ltd', 'although', 'describe', 'to', 'last', 'it', 'might', 'neither', 'within', 'an', 'sincere', 'four', 'the', 'latter', 'whereby', 'only', 'more', 'whatever', 'had', 'except', 'is', 'every', 'nevertheless', 'which', 'whereas', 'me', 'interest', 'about', 'wherein', 'nine', 'whoever', 'top', 'mostly', 'who', 'find', 'give', 'de', 'call', 'whenever', 'in', 'itself', 'just', 'was', 'mine', 'else', 'if', 'some', 'before', 'yourself', 'un', 'alone', 'back', 'therefore', 'cannot', 'themselves', 'throughout', 'would', 'whole', 'etc', 'whose', 'few', 'ever', 'former', 'thi', 'where', 'amongst', 'hence', 'formerly', 'thin', 'serious', 'bottom', 'here', 'after', 'along', 'please', 'as', 'inc', 'anywhere', 'whereupon', 'because', 'whether', 'fifty', 'put', 'up', 'move', 'rather', 'without', 'himself', 'own', 'hundred', 'very', 'someone', 'behind', 'there', 'moreover', 'others', 'said', 'from', 'by', 'therein', 'co', 'three', 'through', 'five', 'why', 'such', 'latterly', 'when', 'sometimes', 'however', 'dont', 'fire', 'nobody', 'down', 'also', 'afterwards', 'both', 'everything', 'besides', 'same', 'indeed', 'her', 'meanwhile', 'amount', 'he', 'whereafter', 'how', 'yet', 'thick', 'expand', 'never', 'these', 'being', 'like', 'towards', 'ourselves', 'done', 'via', 'nowhere', 'of', 'yours', 'but', 'are', 'anything', 'side', 'twenty', 'almost', 'even', 'twelve', 'often', 'over', 'seem', 'their', 'elsewhere', 'them', 'among', 'none', 'thereby', 'thru', 'seemed', 'thence', 'get', 'now', 'something', 'amoungst', 'across', 'several', 'i', 'name', 'could', 'hereafter', 'toward', 'beforehand', 'always', 'off', 'around', 'whence', 'us', 'this', 'with', 'ours', 'hasnt', 'whither', 'wa', 'whom', 'somehow', 'for', 'next', 'him', 'again', 'two', 'upon', 'your', 'cant', 'take', 'be', 'less', 'perhaps', 'while', 'am', 'thus', 'seems', 'thereupon', 'any', 'everywhere', 'or', 'so', 'myself', 'has', 'eight', 'im', 'should', 'must', 'beyond', 'on', 'third', 'they', 'anyone', 'been', 'she', 'still', 'since', 'fill', 'already', 'show', 'per', 'that', 'its', 'nor', 'and', 'namely', 'each', 'hereby', 'cry', 'many', 'during', 'sixty', 'ten', 'have', 'go', 'a', 'may', 'hers', 'nothing', 'most', 'keep', 'empty', 'until', 'anyway', 'another',

'at', 'anyhow', 'all', 'becoming', 'wherever', 'eg', 'than', 'otherwise', 'much', 'will', 'click', 'ie', 'further', 'we', 'those', 'becomes', 'onto', 'under', 'couldnt', 'part', 'out', 'what', 'thing', 'became', 'con', 'my', 'too', 'other', 'you', 'below', 'hereupon', 'sometime', 'least', 'either', 'eleven', 'mill', 'system', 'one', 'together', 'enough', 'against', 'into', 're', 'detail', 'noone', 'thereafter', 'somewhere', 'forty', 'above', 'herself', 'were', 'front', 'not', 'between', 'then', 'no', 'fifteen', 'found', 'see', 'made', 'beside', 'full', 'well', 'his', 'our', 'do', 'yourselves', 'first', 'due', 'once', 'bill', 'herein', 'become', 'everyone', 'can', 'six', 'though']

#7.1 Create vectorizer with combined stop words

vectorizer = CountVectorizer(stop_words='english') # Start with English stopwords

full_stopwords = set(vectorizer.get_stop_words()).union(set(custom_stopwords))

vectorizer = CountVectorizer(stop_words=custom_stopwords)

#7.2 Create document-term matrix

dtm = vectorizer.fit_transform(data['stemmed_comment'])

 # 8. Fit the LDA model

lda_model = LatentDirichletAllocation(n_components=3, random_state=42) # You can adjust n_components (number of topics)

lda_model.fit(dtm)

#8.1 Function to display the top words for each topic

```
def display_topics(model, feature_names, no_top_words=10):
    for topic_idx, topic in enumerate(model.components_):
        print(f"Topic #{topic_idx + 1}:")
        print(" ".join([feature_names[i] for i in topic.argsort()[:-no_top_words - 1:-1]]))
        print("\n")
```

#8.2 Get feature names (words) from the vectorizer

feature_names = vectorizer.get_feature_names_out()

Display the top 10 words for each topic

display_topics(lda_model, feature_names)

#8.3 Assign dominant topic to each comment

topic_distributions = lda_model.transform(dtm)

dominant_topics = np.argmax(topic_distributions, axis=1)

data['dominant_topic'] = dominant_topics

 # 8.4 Assign the most probable topic index to each comment

dominant_topics = topic_distributions.argmax(axis=1)

Add it to your dataframe

data['dominant_topic'] = dominant_topics

 # 8.5Top words for each topic

```
topic_keywords = [
    ['time', 'diabet', 'thi', 'need', 'test', 'day', 'diet', 'insulin', 'carb', 'wa'],  # Topic 1
    ['thing', 'help', 'time', 'just', 'glucos', 'thi', 'wa', 'diabet', 'blood', 'stress'],  # Topic 2
    ['time', 'like', 'work', 'type', 'high', 'click', 'expand', 'need', 'said', 'feel'],  # Topic 3
    ['level', 'just', 'click', 'expand', 'blood', 'im', 'said', 'diabet', 'thi', 'wa']   # Topic 4
]
```

#8.6 Assign labels based on the keywords

```
topic_labels = [
    "Stress Management",
    "Blood Sugar and Stress",
```

```
    "Blood Sugar Control"
]
#8.7 Map the dominant topic to the corresponding label
data['topic_label'] = data['dominant_topic'].apply(lambda x: topic_labels[x])
print(data[['comment', 'dominant_topic', 'topic_label']].head())
#8.8 Count the occurrences of each dominant topic
topic_counts = data['dominant_topic'].value_counts()
#8.9 Display the topic counts
print(topic_counts)
import matplotlib.pyplot as plt
import seaborn as sns
data['dominant_topic_label'] = data['dominant_topic'].map(lambda x:
topic_labels[x])

# 8.10Map the dominant topic numbers to their respective labels
data_unique['dominant_topic_label'] =
data_unique['dominant_topic'].map(lambda x: topic_labels[x])
# 8.11Count of topics before removing duplicates
print("Topic counts before removing duplicates:")
print(data['dominant_topic_label'].value_counts())
# Remove duplicates based on the 'cleaned_comment' column
data_unique = data.drop_duplicates(subset=['cleaned_comment'])
# Count of topics after removing duplicates
print("\nTopic counts after removing duplicates:")
print(data_unique['dominant_topic_label'].value_counts())
#9. coherence test for the model
from gensim.models import CoherenceModel
```

```
from gensim.corpora import Dictionary
#9.1 Extract topics from sklearn LDA model
def sklearn_lda_to_gensim_topics(sklearn_model, vectorizer, topn=10):
    words = vectorizer.get_feature_names_out()
    topics = []
    for topic_weights in sklearn_model.components_:
        top_indices = topic_weights.argsort()[::-1][:topn]
        topic_terms = [(words[i], topic_weights[i]) for i in top_indices]
        topics.append(topic_terms)
#9.2 Convert sklearn topics to gensim format
topics_gensim = [[word for word, _ in topic] for topic in
sklearn_lda_to_gensim_topics(lda_model, vectorizer, topn=10)]
#9.3 Compute Coherence Score using Gensim
coherence_model_lda = CoherenceModel(
    topics=topics_gensim,
    texts=processed_texts,
    dictionary=dictionary,
    coherence='c_v'
)
coherence_lda = coherence_model_lda.get_coherence()
print(f"\n Coherence Score (c_v): {coherence_lda:.4f}")
#9.4 Plot the distribution of dominant topics with their labels
plt.figure(figsize=(10, 6))
sns.countplot(data=data_unique, x='dominant_topic_label', palette='Set2')
plt.title('Distribution of Dominant Topics')
plt.xlabel('Topic')
```

```
plt.ylabel('Number of Comments')
plt.xticks(rotation=45) # Rotate the x labels for better readability
plt.show()
```

 # 10. Function to calculate TextBlob polarity and subjectivity

```
def get_textblob_sentiment(text):
    blob = TextBlob(text)
    return blob.sentiment.polarity, blob.sentiment.subjectivity
```

 # 10.1Apply the function to the 'cleaned_comment' column

```
data[['textblob_polarity', 'textblob_subjectivity']] =
data['cleaned_comment'].apply(lambda x: pd.Series(get_textblob_sentiment(x)))
```

#10.2 Display the first few rows

```
data[['cleaned_comment','textblob_polarity', 'textblob_subjectivity']].head()
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
```

#11. Initialize VADER Sentiment Analyzer

```
analyzer = SentimentIntensityAnalyzer()
```

#11.1 Function to calculate VADER sentiment

```
def get_vader_sentiment(text):
    sentiment = analyzer.polarity_scores(text)
    return sentiment['compound']  # The compound score gives an overall
sentiment
```

#11.2 Apply the function to the 'cleaned_comment' column

```
data['vader_sentiment'] =
data['cleaned_comment'].apply(get_vader_sentiment)
```

#11.3 Display the sentiment results from both TextBlob and VADER

```
data[['cleaned_comment', 'textblob_polarity', 'textblob_subjectivity',
'vader_sentiment']].head()
```

#11.3 Plot polarity distribution

```
plt.figure(figsize=(10,6))
sns.histplot(data['textblob_polarity'], kde=True, color='blue')
plt.title('TextBlob Polarity Distribution')
plt.xlabel('Polarity')
plt.ylabel('Frequency')
plt.show()
```

#11.4 Plot subjectivity distribution

```
plt.figure(figsize=(10,6))
sns.histplot(data['textblob_subjectivity'], kde=True, color='green')
plt.title('TextBlob Subjectivity Distribution')
plt.xlabel('Subjectivity')
plt.ylabel('Frequency')
plt.show()
```

 # 11.5 Plot VADER sentiment distribution

```
plt.figure(figsize=(10,6))
sns.histplot(data['vader_sentiment'], kde=True, color='red')
plt.title('VADER Sentiment Distribution')
plt.xlabel('Sentiment')
plt.ylabel('Frequency')
plt.show()
```

#11.6 Group by topic label and calculate average sentiment scores

```
topic_sentiment = data.groupby('topic_label')[['textblob_polarity',
'textblob_subjectivity', 'vader_sentiment']].mean()
```

#11.7 Display the results

```
topic_sentiment
```

#11.7 Plotting the sentiment scores for each topic

```
plt.figure(figsize=(10,6))
```

#11.8 Plot TextBlob polarity and VADER sentiment for each topic

```
sns.barplot(x=topic_sentiment.index, y=topic_sentiment['textblob_polarity'], color='blue', label='TextBlob Polarity')
sns.barplot(x=topic_sentiment.index, y=topic_sentiment['vader_sentiment'], color='red', label='VADER Sentiment')
plt.title('Average Sentiment by Topic')
plt.xlabel('Topic')
plt.ylabel('Sentiment Score')
plt.xticks(rotation=45)
plt.legend()
plt.show()
```

#12. sentiment analysis VADER

```
def classify_vader(score):
    if score >= 0.05:
        return "positive"
    elif score <= -0.05:
        return "negative"
    else:
        return "neutral"
data['vader_label'] = data['vader_sentiment'].apply(classify_vader)
```

#13. sentiment analysis TextBlob

```
def classify_textblob(score):
    if score > 0.1:
        return "positive"
    elif score < -0.1:
```

```
        return "negative"
    else:
        return "neutral"
data['textblob_label'] = data['textblob_polarity'].apply(classify_textblob)
```

#14.calculate feeling- TextBlob

```
textblob_counts = data['textblob_label'].value_counts()
print("TextBlob Sentiment Counts:")
print(textblob_counts)
```

#15 calculate feelings VADER

```
vader_counts = data['vader_label'].value_counts()
print("\nVADER Sentiment Counts:")
print(vader_counts)
print("all comments before duplicate):", data.shape[0])
print("all comments after duplication:", data_unique.shape[0])
print(data_unique.columns)
from textblob import TextBlob
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
analyzer = SentimentIntensityAnalyzer()
```

 # 16.TextBlob sentiment classification

```
def classify_textblob(comment):
    polarity = TextBlob(comment).sentiment.polarity
    if polarity > 0.1:
        return 'positive'
    elif polarity < -0.1:
        return 'negative'
```

```
        else:
            return 'neutral'
        # cleaned_comment (strings) NaN
    data_unique['cleaned_comment']= data_unique['cleaned_comment'].fillna('')
    data_unique['cleaned_comment’] =
data_unique['cleaned_comment'].astype(str)
    data_unique.loc[:,'textblob_label']=
data_unique['cleaned_comment'].apply(classify_textblob)
```

#17. VADER sentiment classification

```
    def classify_vader(comment):
        score = analyzer.polarity_scores(comment)['compound']
        if score >= 0.05:
            return 'positive'
        elif score <= -0.05:
            return 'negative'
        else:
            return 'neutral'
    data_unique['vader_label'] =
data_unique['cleaned_comment'].apply(classify_vader)
    print("TextBlob Sentiment Counts (271 comment):")
    print(data_unique['textblob_label'].value_counts())
    print("\nVADER Sentiment Counts (271comment):")
    print(data_unique['vader_label'].value_counts())
    from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
    analyzer = SentimentIntensityAnalyzer()
```

```
data_unique['vader_compound']=
data_unique['cleaned_comment'].apply(lambdax:
analyzer.polarity_scores(str(x))['compound'])

from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

analyzer = SentimentIntensityAnalyzer()

data_unique['vader_compound']=
data_unique['cleaned_comment'].apply(lambdax:
analyzer.polarity_scores(str(x))['compound'])

print(data_unique[['textblob_polarity', 'textblob_subjectivity']].head())


import matplotlib.pyplot as plt

# assume comments number are  271

posts = range(len(data_unique))

plt.figure(figsize=(16, 4))

#18. Plot 1 - TextBlob Polarity

plt.subplot(1, 3, 1)

plt.scatter(posts, data_unique['textblob_polarity'], color='lime')

plt.title('Polarity (TextBlob)')

plt.xlabel('Posts')

plt.ylabel('Polarity')

# 19.Plot 2 - TextBlob Subjectivity

plt.subplot(1, 3, 2)

plt.scatter(posts, data_unique['textblob_subjectivity'], color='lime')

plt.title('Sensitivity (TextBlob)')

plt.xlabel('Posts')

plt.ylabel('Sensitivity')
```

```
# 20.Plot 3 - VADER Compound
plt.subplot(1, 3, 3)
plt.scatter(posts, data_unique['vader_compound'], color='lime')
plt.title('Compound Sentiment Score (Vader)')
plt.xlabel('Posts')
plt.ylabel('Sentiment')
plt.tight_layout()
plt.show()
print(data_unique.columns)


print(len(data_unique))
print(data_unique[['textblob_polarity','textblob_subjectivity',
'vader_compound']].count())
from scipy.stats import skew, kurtosis
# 21.Shapiro-Wilk Test
print("Shapiro-Wilk Test:")
for col in ['textblob_polarity', 'textblob_subjectivity', 'vader_compound']:
    stat, p = shapiro(data_unique[col])
    print(f"{col}: W={stat:.4f}, p-value={p:.4f}")
#21.1 Kolmogorov-Smirnov Test
print("\nKolmogorov-Smirnov Test:")
for col in ['textblob_polarity', 'textblob_subjectivity', 'vader_compound']:
    stat, p = kstest(data_unique[col], 'norm')
    print(f"{col}: KS={stat:.4f}, p-value={p:.4f}")
#22. Scatterplot: TextBlob Polarity vs VADER Compound
plt.figure(figsize=(6, 4))
```

```
sns.scatterplot(x='textblob_polarity', y='vader_compound', data=data_unique, color='brown')
plt.title("Figure 3. Scatterplot of Correlation")
plt.xlabel("Polarity (TextBlob)")
plt.ylabel("Compound Sentiment Score (Vader)")
plt.tight_layout()
plt.show()
```

 # 23.Bean plot (violin plot) for polarity, subjectivity, and compound

```
melted_data = pd.melt(
    data_unique[['textblob_polarity', 'textblob_subjectivity', 'vader_compound']],
    var_name='Metric',    value_name='Value'
plt.figure(figsize=(8, 6))
sns.violinplot(x='Metric', y='Value', data=melted_data, inner='box', palette='Greens')
plt.title("Figure 4. Bean Plots of Results")
plt.tight_layout()
plt.show()
# filter comments
filtered_data = data_unique[
   (data_unique['textblob_label'] != 'neutral') &
   (data_unique['vader_label'] != 'neutral')
]
# check if its positive or negative
labels = ['negative', 'positive']
# textblob
```

```
y_true = filtered_data['textblob_label']
y_pred_textblob = filtered_data['textblob_label']
y_pred_vader = filtered_data['vader_label']
from sklearn.metrics import precision_recall_fscore_support
table = []
# TextBlob
textblob_scores = precision_recall_fscore_support(y_true, y_pred_textblob, labels=labels)
for i, label in enumerate(labels):
    table.append({
        'Model': 'TextBlob',
        'Description': label.capitalize(),
        'Precision': round(textblob_scores[0][i], 2),
        'Recall': round(textblob_scores[1][i], 2),
        'F1-Score': round(textblob_scores[2][i], 2),
        'Support': textblob_scores[3][i]
    })
table.append({
    'Model': 'TextBlob',
    'Description': 'Accuracy',
    'Precision': '',
    'Recall': '',
    'F1-Score': round((textblob_scores[2][0] + textblob_scores[2][1]) / 2, 2),
    'Support': sum(textblob_scores[3])
})
# VADER
```

```
        vader_scores = precision_recall_fscore_support(y_true, y_pred_vader,
labels=labels)
        for i, label in enumerate(labels):
            table.append({
                'Model': 'VADER',
                'Description': label.capitalize(),
                'Precision': round(vader_scores[0][i], 2),
                'Recall': round(vader_scores[1][i], 2),
                'F1-Score': round(vader_scores[2][i], 2),
                'Support': vader_scores[3][i]
            })
        table.append({
            'Model': 'VADER',
            'Description': 'Accuracy',
            'Precision': '',
            'Recall': '',
            'F1-Score': round((vader_scores[2][0] + vader_scores[2][1]) / 2, 2),
            'Support': sum(vader_scores[3])
        })
        # no empty spaces in textblob_polarity
        count_polarity = data_unique['textblob_polarity'].notna().sum()
        print(f"Number of comments with TextBlob polarity: {count_polarity}")
        data_unique.groupby('dominant_topic_label')['vader_compound'].mean().sort
_values(ascending=False)
        df = pd.DataFrame({
            'textblob_polarity': np.random.uniform(-1, 1, 1000),
            'textblob_subjectivity': np.random.uniform(0, 1, 1000),
```

```
    'vader_compound': np.random.uniform(-1, 1, 1000)
})
def describe_sentiment(column):
    return {
        'Mean': np.mean(column),
        'Median': np.median(column),
        'Std. Dev': np.std(column, ddof=1),
        'Range': np.max(column) - np.min(column),
        'IQR': np.percentile(column, 75) - np.percentile(column, 25),
        'Skewness': skew(column),
        'Kurtosis': kurtosis(column)
    }
# statistcs calculation
results = {
    'TextBlob Polarity': describe_sentiment(df['textblob_polarity']),
    'TextBlob Subjectivity': describe_sentiment(df['textblob_subjectivity']),
    'VADER Compound': describe_sentiment(df['vader_compound'])
}
```

RESUME

Engy Mohamed Khalil ALI

Profile Summary

Dynamic and ambitious pharmaceutical professional with over 10 years of experience in medical sales, import/export, and international consultancy. Strong interpersonal skills, with proven ability to work in diverse cultural environments. Fluent in Arabic and English, with working proficiency in Turkish and German.

Key Skills:

- Teamwork & Collaboration
- Leadership & Initiative
- Ability to Work Under Pressure
- Problem Solving
- Strong Communication
- Hardworking & Supportive
- Presentable & Dynamic
- Self-Motivated & Goal-Oriented

Education:

- *Bachelor of Pharmaceutical Sciences*Misr International University (MIU), Egypt 2006 – 2011

Professional Experience:

- Medical Representative Hygint Pharma – Egypt January 2013 – October 2015

  - Promoted pharmaceutical products to healthcare professionals.

  - Built and maintained strong customer relationships.

  - Increased brand awareness and expanded client base.

- Freelance Import & Export Specialist November 2015 – December 2019

- Facilitated international trade of medicines, medical supplies, and supplements.

- Collaborated with Egyptian companies to meet demand for imported medical products.

- Career Break December 2019 – May 2021

  - Returned to Egypt due to the passing of my father.

  - Quarantined during the COVID-19 pandemic in Turkey.

- Medical Consultant (International Patients) August 2021 – July 2023

  - Advised international patients on treatment options including plastic surgery, hair transplants, and dental procedures.

  - Managed sales and patient coordination for medical tourism services.

- Master student at Gedik university July 2023 -now

  Languages:

  Arabic – Native

  English – Fluent

  Turkish – Intermediate (B1)

  German – Basic (A2)