ANALYZING READER NEEDS THROUGH SELF-HELP BOOK RECOMMENDATIONS

By:

Cordero, Ricci Ayn D.

Zamoras, Stephanie M.

A Technical Project submitted to the Mapua University, School of Information Technology in Partial Fulfillment of the Requirements for

ITS132L: Data Warehousing and Data Mining

Mapúa University

August 2025

Table of Contents

1. Introduction

1.1. Overview of the Project

This project analyzes how users in Reddit's r/booksuggestions community seek and recommend self-help books. By examining these interactions, the study aims to uncover how book trends correlate with underlying reader needs reflected in recommendation requests. As interest in mental health and personal development continues to grow, understanding these digital conversations provides valuable insights into behavioral patterns.

Approximately 6,000 Reddit posts were collected using targeted keywords such as “self-help,” “motivation,” “anxiety,” and “mental health.” These terms represent common psychological and emotional challenges faced by users. Each post was extracted along with its metadata—such as title, content, comments, and engagement metrics—making the dataset suitable for text mining, sentiment analysis, and thematic exploration.

The project demonstrates the practical application of data warehousing and mining concepts in handling unstructured data. It addresses real-world challenges such as API rate limits, noisy or incomplete text, and the need for effective data transformation. Posts and comments were structured into CSV and JSON formats, enabling advanced analytical tasks like keyword clustering, engagement trend detection, and sentiment scoring.

By focusing on self-help content, this study highlights how online communities turn to literature for support and self-improvement. It contributes to a better understanding of digital reader behavior and offers practical implications for recommender systems, content creators, and mental wellness advocates.

1.1. Objectives and significance

This project aims to uncover patterns in how users on Reddit’s r/booksuggestions subreddit seek and recommend self-help literature. By examining the language, tone, and engagement patterns of these posts, the study explores how emotional needs and personal challenges are reflected in book recommendations shared online.

1.2.1 Main Objectives

The primary objective of this project is to analyze user-generated content from Reddit’s r/booksuggestions to identify behavioral patterns and emerging emotional needs among readers interested in self-help literature. By converting unstructured

Reddit data into structured formats, the study applies data warehousing and mining techniques to generate actionable insights on user concerns, interests, and recommendation trends.

1.2.2 Sub-Objectives

To support the main goal, the study is guided by the following sub-objectives:

- **Identify popular self-help themes -** Determine which psychological or motivational topics (e.g., anxiety, productivity, discipline) are most frequently mentioned, based on keyword occurrence and discussion volume.
- **Extract engagement patterns -** Analyze metrics such as upvotes and comment counts to identify which types of posts generate the most interaction and support within the community.
- **Assess sentiment and user intent -** Apply sentiment analysis to classify emotional tone in posts and comments, and group content into themes such as seeking advice, sharing experiences, or offering recommendations.

1.2.3 Broader Importance

The significance of this project extends across academic, social, and commercial domains:

Target Users or Audience

The results of this study are relevant to multiple stakeholders, including:

- **Online Book Retailers and Recommendation Platforms -** These platforms can use the findings to improve recommendation engines, tailoring suggestions based on users' emotional needs and trending psychological topics.
- **Mental Health Advocates and Support Groups -** Insights into emotional themes and user sentiment can inform outreach programs and mental wellness initiatives by identifying pressing concerns within online communities.
- **Academic Researchers in Social Behavior and NLP -** The structured dataset enables further research into digital behavior, emotional expression, and the linguistic markers of help-seeking behavior in online forums.
- **Content Creators and Self-Help Authors -** Writers and influencers can respond to user demand by addressing underrepresented topics or enhancing relevance through emotionally aligned content.

These audiences were selected due to their direct involvement in behavioral insights, personalized content delivery, and the growing intersection of mental wellness and digital media.

1.2. Target users or audience

This study identifies several stakeholders who could benefit from the insights extracted from the Reddit self-help book recommendation dataset. By understanding user sentiment, engagement, and thematic preferences, these groups can make informed decisions in their respective fields.

- **Online Book Retailers and Recommendation Platforms -** Platforms such as Amazon or Goodreads can enhance their recommendation algorithms using patterns derived from trending emotional and psychological needs. This enables more personalized and emotionally resonant book suggestions, improving user satisfaction and retention.
- **Mental Health Advocates and Support Organizations -** The emotional tone of posts and comments can help identify common struggles within online communities. Advocacy groups can use these insights to design targeted outreach campaigns or develop supportive resources tailored to the most frequently expressed needs.
- **Academic Researchers in Social Behavior and Natural Language Processing (NLP) -** The structured dataset offers a valuable foundation for studies in digital behavior, emotional expression, and community dynamics. Researchers in psychology, linguistics, and data science can explore correlations between language and emotional well-being, or test models for sentiment classification.
- **Content Creators and Self-Help Authors -** Writers, bloggers, and influencers producing self-help content can use this data to identify in-demand topics, emotional themes, or underserved niches. This allows for more relevant and impactful content creation that resonates with readers' lived experiences.

These audiences were selected based on their vested interest in behavioral insights, content personalization, and the growing public reliance on digital communities for mental and emotional support.

2. Data Source and Collection

2.1. Description of the unstructured data source

The primary data source for this project is Reddit, specifically the r/booksuggestions subreddit. This online community provides a platform where users seek and share book recommendations, often centered around personal concerns or emotional needs. The unstructured data extracted from this subreddit includes:

- Post titles and body content
- Number of upvotes and comments
- Keyword used to retrieve the post
- Comments under each post, including replies, upvotes, and sentiment scores

The dataset focuses on the self-help and mental wellness domain, covering topics such as anxiety, confidence, motivation, trauma, discipline, and meditation. Each post and its associated comments reflect real user concerns, providing insight into emotional triggers and book-based coping strategies.

Limitations of Reddit as a data source:

- Some posts are deleted or edited after scraping.
- Text quality is inconsistent due to informal grammar, slang, and emojis.
- Duplicate or similar discussions may appear across multiple keyword searches.

All extracted data was stored in CSV and JSON formats for structured processing and sentiment analysis.

2.2. Scraping Tools and Libraries Used

The following tools and Python libraries were used for data extraction and verification:

- **PRAW (Python Reddit API Wrapper):** Used for collecting posts and comments. PRAW simplifies Reddit API access, enabling efficient filtering, metadata extraction, and iteration over results.
- **TextBlob:** Used for sentiment analysis on post titles and comment content. Sentiment polarity scores range from –1.0 (very negative) to 1.0 (very positive).
- **Pandas:** Used to structure the data into DataFrames and export to .csv.

- **JSON:** Used to format nested comment data into a readable structure.
- **Tqdm:** Adds progress bars to monitor scraping progress.
- **Pathlib and OrderedDict:** Used for file handling and preserving data order.

Justification:

PRAW was chosen over alternatives like Selenium or BeautifulSoup because it interacts directly with Reddit's API, avoiding issues with dynamic page content. TextBlob was selected due to its simplicity and efficiency for sentiment scoring without requiring model training.

2.3. Scraping Strategy and Logic

The scraping strategy was **keyword-driven**, with each keyword representing a specific self-help theme (e.g., "motivation," "discipline," "mental health").

Authentication and Setup

- Reddit API access was authorized using OAuth2 credentials (client_id, client_secret, and user_agent) via PRAW.

Scraping Process

1. A curated list of ~30 self-help-related keywords was created.
2. For each keyword:
    - Query r/booksuggestions using subreddit.search() with sort="new" and a post limit to avoid rate limiting.
    - Extract metadata: post ID, title, content (selftext), upvotes and comment count.
    - Retrieve top-level comments and analyze their sentiment using TextBlob.
    - Store both post and comment data, ensuring no duplication by tracking unique post IDs.

Simplified Pseudocode

```
for keyword in keywords:
    posts = subreddit.search(keyword, sort="new", limit=200)
    for post in posts:
        extract post fields
```

for comment in post.comments:

apply TextBlob sentiment

store comment data

Notes

- Sentiment scores were captured for both post titles and comment texts.
- Duplicates were prevented using unique post IDs.
- Delays and retries were implemented to comply with Reddit API limits.

2.4. Data Acquisition Timeline

Phase	Date Started	Date Completed	Duration	Notes
Keyword Curation	July 18, 2025	July 18, 2025	1 day	Finalized a list of 30+ self-help-related keywords.
Test Scraping & Debugging	July 19, 2025	July 19, 2025	1 day	Verified API access and comment structure.
Full Scraping Run	July 20, 2025	July 21, 2025	2 days	Collected approximately 6,000 posts.
Comment Extraction	July 22, 2025	July 23, 2025	2 days	Created nested JSON file and applied sentiment analysis
Post-Processing & Cleanup	July 24, 2025	July 25, 2025	2 days	Exported cleaned CSV/JSON files and removed duplicates.

No significant interruptions occurred during the data acquisition process. Posts that were deleted or removed before scraping were automatically skipped. The final dataset includes both post-level and comment-level sentiment scoring.

3. Data Preprocessing

3.1. Data Cleaning Techniques

The raw dataset extracted from Reddit included several inconsistencies and unwanted entries. These issues needed to be addressed to ensure reliable downstream analysis. The following data cleaning strategies were implemented:

Issue Identified	Cleaning Action Applied	Before	After
Missing post titles or bodies	Removed posts with null or empty titles or body content	6,000	5,953
Deleted or removed comments/posts	Skipped posts marked as [deleted] or [removed] during extraction		Excluded
Duplicate post IDs	Checked and removed duplicates using unique post_id as identifier	6,000	5,953
Inconsistent text formatting	Removed extra whitespaces, fixed escape characters (e.g., \n, \")		Cleaned

Cleaning was primarily done using Pandas, and post IDs were used to ensure uniqueness. Additionally, only English posts with least some engagement (e.g., non-zero comments or upvotes) were retained.

3.2. Parsing and Normalization

The dataset originally existed in a nested JSON format. Each post object contained multiple fields, including a list of comments, each with its own text, sentiment, and upvotes. To perform meaningful analysis, this hierarchical structure was flattened into a tabular format where each row corresponded to a book mention candidate extracted from a comment.

Parsing Strategy

A multi-step text extraction pipeline was implemented to identify book titles mentioned in comments. Three main patterns were used via regular expressions:

1. Quoted Titles: Matches book names enclosed in quotes, e.g., "The Power of Now".
2. "by Author" Patterns: Captures the book title in phrases like Atomic Habits by James Clear.
3. Capitalized Phrases: Captures likely titles based on consecutive capitalized words, e.g., The Subtle Art of Not Giving a F*ck.

A Python function called harvest_titles(text) applied these patterns to comment text and returned likely book titles. The function filtered out noise by ignoring very short/long strings or titles with over 10 words.

```
QUOTED = re.compile(r"[\"“”'‘’](.+?)[\"“”'‘’]")
BY_AUTHOR = re.compile(r"(.+?)\s+by\s+[\w\s\.\-']{2,}", re.I)
CAP_TITLE = re.compile(r"\b([A-Z][\w'\-]*(?:\s+[A-Z][\w'\-]*){1,5})")


with open("reddit_booksuggestions_cleaned.json", "r", encoding="utf-8") as
f:
    posts = json.load(f)

def harvest_titles(text):
    titles = set()
    titles.update(QUOTED.findall(text))
    titles.update(BY_AUTHOR.findall(text))
    if not titles:
        titles.update(CAP_TITLE.findall(text))
    return [
        t.strip() for t in titles
        if 2 <= len(t.split()) <= 10 and len(t) <= 100
    ]
```

Flattening the JSON

For each post:

- The top-level keyword and post_id were extracted.
- Each comment was checked for type validity and cleaned.
- The harvest_titles() function was applied to extract possible book titles.
- Each candidate title was stored with associated metadata: sentiment, upvotes, comment, and post_index.

This process resulted in a flat list of title candidates for matching and further analysis.

Normalization

To ensure consistency and enable downstream joining or filtering, the following normalization was applied:

Normalization Step	Description
title_raw filtering	Removed overly short or overly long entries, kept 2-10 word titles
Text stripping	Removed leading/trailing whitespace in comment text and titles
Categorical mapping (post data)	Preserved and standardized keyword, post_id, and post_index fields
Encoding consistency	Saved final outputs in UTF-8 (JSON and CSV) for broad compatibility

Matching and Verification

Extracted title_raw values were matched to a known book title dataset (books.csv) using RapidFuzz fuzzy matching. Only titles with a similarity score ≥ 90 were accepted as verified.

```
verified = []
for entry in tqdm(candidates, desc="Offline Match"):
    match, score, _ = process.extractOne(entry["title_raw"], known_titles)
    if score >= 90:  # Match threshold
        entry["verified_title"] = match
        entry["match_score"] = score
        verified.append(entry)
```

Output Format

The cleaned and verified dataset was exported in two formats:

- verified_books_offline.csv: Tabular format for analysis.
- verified_books_offline.json: JSON for flexible use or further processing.

These files include the following fields:

- title_raw
- verified_title
- match_score
- comment
- sentiment

- upvotes
- keyword
- post_index
- post_id

This output provides a clean foundation for future mining, visualization, or clustering steps.

3.3. Data transformation

The following transformation were applied to ensure the dataset was suitable for analysis and visualization:

Transformation	Description	Reason
Title Conversion	Stripped whitespace from post titles (post["title"].strip()).	Ensures consistent formatting for analysis.
Keyword standardization	Converted keywords to lowercase and trimmed whitespace (post["keyword"].strip().lower()).	Facilitates accurate grouping and filtering by keyword.
Comment Validation	Kept only comments with ≥10 characters, excluding [deleted] or [removed].	Removes low-quality or irrelevant text.
Deduplication	Removed duplicate comments using a seen_texts set.	Eliminates redundant entries for cleaner analysis.
Field Standardization	Retained only comment_id, text, upvotes, and sentiment in comments.	Simplifies structure and ensures key fields are populated.
Post Filtering	Dropped posts with no valid comments and added num_comments count.	Guarantees usable data for downstream tasks (e.g., sentiment analysis).

Purpose

These transformations ensure the dataset is **structured, deduplicated, and free of noise**, making it suitable for:

- Sentiment analysis (via sentiment scores).
- Trend analysis (e.g., popular keywords).

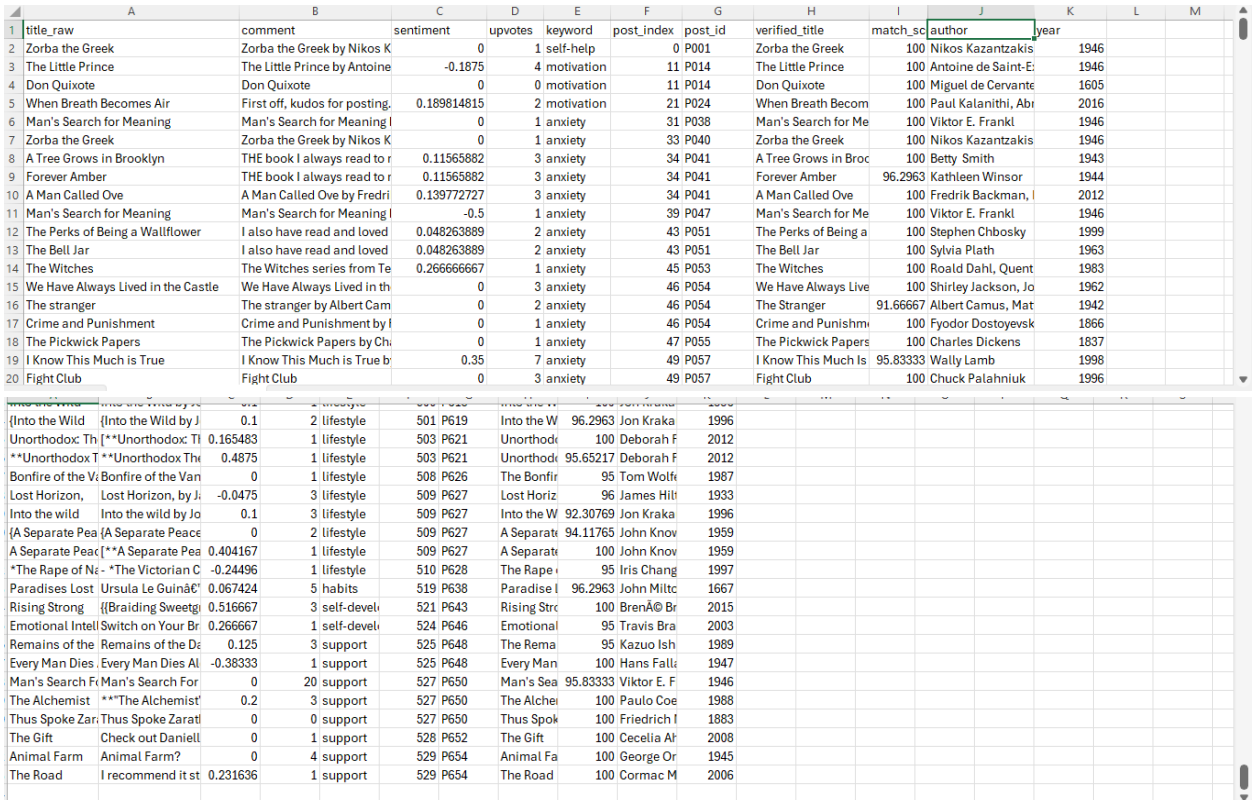
- NLP tasks (clean text fields).

Output: Cleaned JSON file (reddit_booksuggestions_cleaned.json) with standardized fields.

3.4. Sample records or schema after Preprocessing

The dataset is now structured, with false positives removed and inconsistencies resolved, making it suitable for warehousing and data mining applications, which in this case, will be Power BI. Key preprocessing steps included standardizing fields such as "title_raw" and "verified_title," consolidating author names, and cleaning numerical values like "sentiment" and "year." Sentiment scores will be analyzed to identify trends, while clustering will group similar posts based on keywords or other attributes. This cleaned and organized dataset is now optimized for further exploration and analysis.


```
  {
    "title_raw": "Zorba the Greek",
    "comment": "Zorba the Greek by Nikos Kazantzakis",
    "sentiment": 0.0,
    "upvotes": 1,
    "keyword": "self-help",
    "post_index": 0,
    "post_id": "P001",
    "verified_title": "Zorba the Greek",
    "match_score": 100.0,
    "author": "Nikos Kazantzakis, Νίκος Καζαντζάκης",
    "year": 1946.0
  },
  {
    "title_raw": "The Little Prince",
    "comment": "The Little Prince by Antoine de Saint-Exupéry  The Tao
of Pooh by Benjamin Hoff",
    "sentiment": -0.1875,
    "upvotes": 4,
    "keyword": "motivation",
    "post_index": 11,
    "post_id": "P014",
    "verified_title": "The Little Prince",
    "match_score": 100.0,
    "author": "Antoine de Saint-Exupéry, Richard Howard, Dom Marcos
Barbosa, Melina Karakosta",
    "year": 1946.0
  },
```



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	title_raw	comment	sentiment	upvotes	keyword	post_index	post_id	verified_title	match_sc	author	year		
2	Zorba the Greek	Zorba the Greek by Nikos K	0	1	self-help	0	P001	Zorba the Greek	100	Nikos Kazantzakis	1946		
3	The Little Prince	The Little Prince by Antoine	-0.1875	4	motivation	11	P014	The Little Prince	100	Antoine de Saint-E:	1946		
4	Don Quixote	Don Quixote	0	0	motivation	11	P014	Don Quixote	100	Miguel de Cervante	1605		
5	When Breath Becomes Air	First off, kudos for posting.	0.189814815	2	motivation	21	P024	When Breath Becom	100	Paul Kalanithi, Abı	2016		
6	Man's Search for Meaning	Man's Search for Meaning I	0	1	anxiety	31	P038	Man's Search for Me	100	Viktor E. Frankl	1946		
7	Zorba the Greek	Zorba the Greek by Nikos K	0	1	anxiety	33	P040	Zorba the Greek	100	Nikos Kazantzakis	1946		
8	A Tree Grows in Brooklyn	THE book I always read to r	0.11565882	3	anxiety	34	P041	A Tree Grows in Broc	100	Betty Smith	1943		
9	Forever Amber	THE book I always read to r	0.11565882	3	anxiety	34	P041	Forever Amber	96.2963	Kathleen Winsor	1944		
10	A Man Called Ove	A Man Called Ove by Fredri	0.139772727	3	anxiety	34	P041	A Man Called Ove	100	Fredrik Backman, I	2012		
11	Man's Search for Meaning	Man's Search for Meaning I	-0.5	1	anxiety	39	P047	Man's Search for Me	100	Viktor E. Frankl	1946		
12	The Perks of Being a Wallflower	I also have read and loved	0.048263889	2	anxiety	43	P051	The Perks of Being a	100	Stephen Chbosky	1999		
13	The Bell Jar	I also have read and loved	0.048263889	2	anxiety	43	P051	The Bell Jar	100	Sylvia Plath	1963		
14	The Witches	The Witches series from Te	0.266666667	1	anxiety	45	P053	The Witches	100	Roald Dahl, Quent	1983		
15	We Have Always Lived in the Castle	We Have Always Lived in th	0	3	anxiety	46	P054	We Have Always Live	100	Shirley Jackson, Jo	1962		
16	The stranger	The stranger by Albert Cam	0	2	anxiety	46	P054	The Stranger	91.66667	Albert Camus, Mat	1942		
17	Crime and Punishment	Crime and Punishment by I	0	1	anxiety	46	P054	Crime and Punishm	100	Fyodor Dostoyevsk	1866		
18	The Pickwick Papers	The Pickwick Papers by Ch	0	1	anxiety	47	P055	The Pickwick Papers	100	Charles Dickens	1837		
19	I Know This Much is True	I Know This Much is True b	0.35	7	anxiety	49	P057	I Know This Much Is	95.83333	Wally Lamb	1998		
20	Fight Club	Fight Club	0	3	anxiety	49	P057	Fight Club	100	Chuck Palahniuk	1996		
	{Into the Wild	{Into the Wild by J	0.1	2	lifestyle	501	P619	Into the W	96.2963	Jon Kraka	1996		
	Unorthodox: Th	[**Unorthodox: Tl	0.165483	1	lifestyle	503	P621	Unorthod	100	Deborah F	2012		
	**Unorthodox T	**Unorthodox The	0.4875	1	lifestyle	503	P621	Unorthod	95.65217	Deborah F	2012		
	Bonfire of the Vi	Bonfire of the Van	0	1	lifestyle	508	P626	The Bonfir	95	Tom Wolfe	1987		
	Lost Horizon,	Lost Horizon, by J	-0.0475	3	lifestyle	509	P627	Lost Horiz	96	James Hilt	1933		
	Into the wild	Into the wild by Jo	0.1	3	lifestyle	509	P627	Into the W	92.30769	Jon Kraka	1996		
	{A Separate Pea	{A Separate Peace	0	2	lifestyle	509	P627	A Separate	94.11765	John Knov	1959		
	A Separate Peac	[**A Separate Pea	0.404167	1	lifestyle	509	P627	A Separate	100	John Knov	1959		
	*The Rape of Na	- *The Victorian C	-0.24496	1	lifestyle	510	P628	The Rape	95	Iris Chang	1997		
	Paradises Lost	Ursula Le Guinâ€	0.067424	5	habits	519	P638	Paradise I	96.2963	John Miltc	1667		
	Rising Strong	{{Braiding Sweetg	0.516667	3	self-develi	521	P643	Rising Strc	100	BrenÃ© Br	2015		
	Emotional Intell	Switch on Your Br	0.266667	1	self-develi	524	P646	Emotional	95	Travis Bra	2003		
	Remains of the	Remains of the Da	0.125	3	support	525	P648	The Rema	95	Kazuo Ish	1989		
	Every Man Dies	Every Man Dies Al	-0.38333	1	support	525	P648	Every Man	100	Hans Falla	1947		
	Man's Search Fo	Man's Search For	0	20	support	527	P650	Man's Sea	95.83333	Viktor E. F	1946		
	The Alchemist	***The Alchemist'	0.2	3	support	527	P650	The Alche	100	Paulo Coe	1988		
	Thus Spoke Zar	Thus Spoke Zarat	0	0	support	527	P650	Thus Spok	100	Friedrich I	1883		
	The Gift	Check out Daniell	0	1	support	528	P652	The Gift	100	Cecelia Ah	2008		
	Animal Farm	Animal Farm?	0	4	support	529	P654	Animal Fa	100	George Or	1945		
	The Road	I recommend it st	0.231636	1	support	529	P654	The Road	100	Cormac M	2006		

4. Data Warehousing Concepts Applied

4.1. Fact and Dimension Tables

The model is centered on a fact table (fact_reviews) that records each instance where a Reddit user recommended or mentioned a self-help book in a comment. This is supported by four-dimension tables that provide contextual information, enabling flexible analysis across multiple perspectives.

Fact Table: fact_reviews

Field	Description
post_id	Unique identifier of the Reddit post where the book mention occurred
post_index	Numerical index of the posts (used for filtering/sorting)
sentiment	Polarity score (range: -1 to +1) of the comment containing the book mention
Sentiment Category	Derived classification: Positive, Neutral, or Negative
title_row	Book titles as extracted from the comment using regex and string parsing

verified_title	Matched and validated title from the reference dataset (books.csv)
upvotes	Upvote count on the comment indicating engagement level
year	Year of publication of the verified book (if available)
Verified Title Count	Aggregated count of how many times the title appeared in the dataset.

Dimension tables

Dimension Table	Key Field	Description
Dim_books	Verified_title	Includes metadata such as author, keyword, year, and associated sentiment
Dim_keywords	keywords	Contains the original query keyword used to retrieve the Reddit post
Dim_sentiment	sentiment	Maps numeric sentiment scores to human-readable categories (sentiment_category)

This structure enables the warehouse to support multidimensional analysi, such as:

- Tracking which books are most commonly recommended under each emotinal keyword.
- Segmenting recommendations by user sentiment.
- Comparing engagement (upvotes) across sentiment categories or keywords.

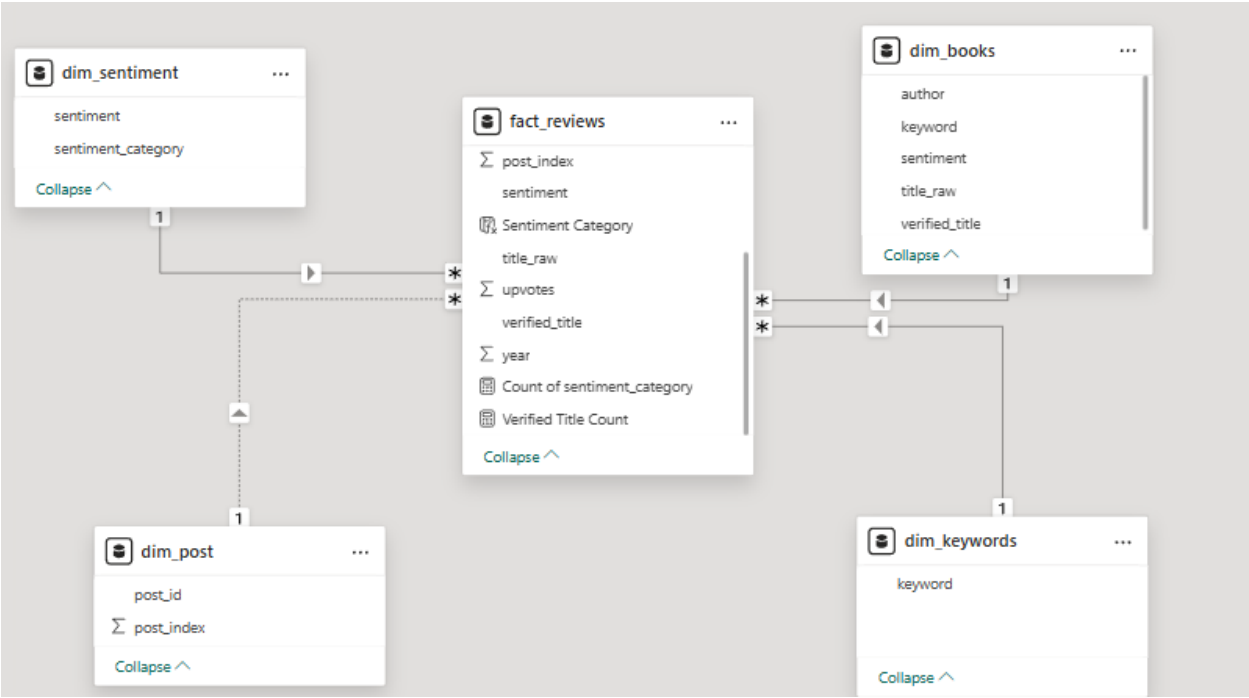
4.2. Schema

The data warehouse follows a star schema, which consists of a central fact table connected to denormalized dimension tables through primary-foreign key relationships. This schema type was selected because of its simplicity, query, efficiency, and compatibility with Power BI's analytical engine.

Justification:

- Star schemas are highly optimized for OLAP-style queries
- Denormalized dimensions tables reduced join complexity and enhance [performance.
- Relationship in Power BI are clearly defined and visually represented in the model view for easy maintenance.

As shown in the Power BI relationship diagram, each dimension table has a 1-to-many relationship with the fact_reviews table, with the foreign keys clearly defined. This structure allows the analyst to filter the fact table across any dimension.



4.3. ETL process description

The data was processed through a well-defined ETL (Extract, Transform, Load) pipeline implement in Python and Power BI. Each phase is described below:

Extract:

- Data was extracted from Reddit's r/booksuggestions subreddit using PRAW (Python Reddit API Wrapper).
- A curated list of 20+ keywords was used to search posts.
- Each Reddit post's title, content, metadata, and up to 100 top-level comments were collected.
- Sentiment scores for each title and comment were computed using TextBlob
- The extracted data was saved to a raw JSON file named reddit_booksuggestion_non_relational.json.

Transform:

- Using clean_reddit_data.py each post was normalized:

    - Whitespace was removed from the fields.
    - Deleted/removed/short comments were filtered out.
    - Duplicates were removed using a seen_texts check
- The result was saved as reddit_booksuggestion_cleaned.json.
- Using extractandverify.py book titles were extracted from comments usning three strategiers:
    1. Title enclosed in quotation marks
    2. Title followed by “by [Author”
    3. Capitalized multi-word phases
- The extracted raw titles were compared to a reference list (book.csv) using RapidFuzz. A similarity score of ≥90 was requred to comfirm a match. Verified titles were saved in:
    - verified_books_offline.csv (for Power BI)
    - verified_books_offline.json (for additional JSON processing)

Load:

- Cleaned and verified datasets were loaded into Power BI as separate tables.
- Relationships were modeled as a star schema, linking fact_reviews to dim_books, dim_keywords, dim_sentiment, and dim_post.
- Calculated columns such as Sentiment Category and Verified Title Count were added for additional analysis.

4.4. Tools used

A combination of open-source tools and commercial platforms was used throughout the pipeline:

Tool/Technology	Role in Project
Python 3.11	Core language for data scraping, transformation, and matching
PRAW (Reddit API)	Used to extract post and comments data from Reddit efficiently
TextBlob	Provided sentiment polarity scores without requiring custom model training
RapidFuzz	Performed fuzzy string matching to link extracted titles to know book names
Pandas	Handled structured tabular transformation, deduplication, export to CSV
JSON/CSV	Used as intermediate formats between scripts and Power BI import

Tqdm/OrderedDict	Enabled progress tracking and structured data formatting during preprocessing
MySQL	Used to normalize tables
PowerBI	Enabled visualization of KPIs using graphs, and the schema.

Why these tools were chosen:

- PRAW and TextBlob provided simple, effective access to Reddit data and sentiment scoring without the overhead of browser automation or machine learning model training.
- RapidFuzz allowed tolerant title verification even with misspellings or formatting inconsistencies.
- Power BI was ideal for building relationships, enabling filtering across sentiment and keywords, and generated user-friendly dashboards,

5. Data Mining Techniques Used

5.1. Sentiment Analysis

To better understand the emotional tone of the self-help book recommendations and reader experiences, the project applied **sentiment analysis** using the TextBlob library. Each comment extracted from Reddit was analyzed for polarity (ranging from -1 for negative to +1 for positive). This allowed the project to categorize user recommendations as **positive, neutral, or negative**, depending on the sentiment score of each comment.

Sample insights gained through sentiment analysis:

- Comments with **motivational** or **uplifting** tones tended to have higher upvotes.
- Neutral or slightly positive sentiments were the most common in book suggestions.
- Negative sentiments were more often associated with personal anecdotes or mental health struggles.

This analysis helped highlight not just which books were recommended, but the **emotional framing** users attached to them.

5.2. Trend Analysis

Trend detection was conducted by grouping Reddit posts by **topic keyword** (e.g., “anxiety,” “motivation”) and analyzing their frequency. The system tracked:

- Most frequently recommended books per keyword
- Popular authors across different emotional themes

This helped identify **reading trends** among users struggling with specific issues, such as:

- *Man’s Search for Meaning* being heavily associated with “anxiety”
- *The Little Prince* and *Don Quixote* showing up under both “self-help” and “motivation”

5.3. Justification for technique selection

The mining techniques used in this project were carefully selected to match both the unstructured nature of the Reddit data and the behavioral insights targeted by the objectives. Each method contributed meaningfully to transforming raw user discussions into structured, analyzable trends.

1. Sentiment Analysis (TextBlob)

- One key objective was to understand the emotional tone behind book recommendations and user experiences. Reddit posts and comments are often personal and emotion-driven, making sentiment analysis an ideal technique.
- TextBlob provides quick and interpretable sentiment polarity scores, enabling classification of emotional tone as positive, neutral, or negative without needing complex model training.
- Results showed that while Reddit posts often had neutral or mixed sentiment, responses were generally supportive and more positive.

2. Fuzzy String Matching (RapidFuzz)

- Comments on Reddit often contain misspelled, informal, or loosely formatted book titles. Exact matching with a book dataset would miss many valid entries.
- RapidFuzz enabled approximate titles matching to a known book list, improving the accuracy of book verification while allowing for real-world text inconsistencies. RapidFuzz helped map raw, user-submitted book titles, often informal or misspelled, onto a verified book list. This allowed for accurate title tracking even

with inconsistent formatting, which also supported the goal of validating recommended books reliably, even when users didn’t quote titles perfectly.

3. Keyword-Driven Thematic Grouping

- The project's core focus was on uncovering self-help themes, so organizing data by these keywords was essential for trend tracking.
- Keyword tagging allowed for trend detection, clustering emotional concerns with frequently recommended books. It allowed posts to be categorized according to emotional needs and psychological topics such as “anxiety,” “confidence,” or “discipline.” This facilitated trend and frequency analysis across user concernsirectly addressed sub-objectives like identifying popular self-help topics and patterns in engagement by theme.

4. Data Structuring with Pandas & JSON

- Why it fits: Reddit’s JSON-based data structure required flattening and transformation for any meaningful analysis.
- Strength: Tools like pandas, json, and OrderedDict enabled efficient transformation into clean CSV/JSON tables, making the data compatible with analytical techniques.
- Impact: Made it possible to apply data mining workflows, sentiment scoring, and trend visualizations across the dataset.

The techniques used in this project balanced simplicity, interpretability, and scalability, making them ideal for:

- Handling noisy, user-generated Reddit data
- Surfacing meaningful insights about reader behavior
- Achieving the objectives in a reproducible, academically sound way

These techniques effectively bridged the gap between unstructured online conversations and structured behavioral insight extraction.

6. Insights and Results

6.1. Summary of Findings

After mining and analyzing approximately 5,953 Reddit posts from the r/booksuggestions subreddit, several significant findings emerged regarding user behavior, preferences, and emotional states tied to self-help literature.

The sentiment analysis showed that most users express neutral to mildly positive tones when requesting self-help book recommendations. Posts tagged with keywords like “motivation,” “discipline,” and “mental health” were among the most frequently used, indicating recurring concerns around productivity and emotional well-being.

Comment analysis revealed that highly recommended books tend to be accompanied by personal stories, emotional relatability, or specific results experienced by the commenter. Recommendations that combined personal context with emotional support received higher upvotes and engagement.

The engagement metrics also confirmed that posts with more emotionally vulnerable titles or specific situations (e.g., “I feel lost in life, any book suggestions?”) received more comments and support than vague or general requests.

6.2. Key Patterns or Anomalies

Key Patterns Identified

- **Peak Posting Times**: Most posts were made on **Sundays and Mondays**, suggesting that users are more reflective or emotionally vulnerable at the start of the week.
- **Top Keywords**: The most frequently occurring keywords were **“anxiety,” “self-help,” “motivation,”** and **“mental health.”** These align with real-world mental health challenges amplified by academic or work stress.
- **Sentiment Skew**: Comment sentiment was generally more positive than post sentiment, implying that Reddit users often respond supportively to those seeking help.

Anomalies Observed

- Some posts containing negative sentiment (e.g., titles about feeling “worthless” or “burned out”) generated **unexpectedly low engagement**. This could be due to discomfort in engaging with highly sensitive topics or platform-specific behavior.

- Certain lesser-known keywords, such as “faith” and “spiritual,” appeared less frequently but received disproportionately high positive sentiment in the responses, suggesting niche interest groups that are highly engaged.

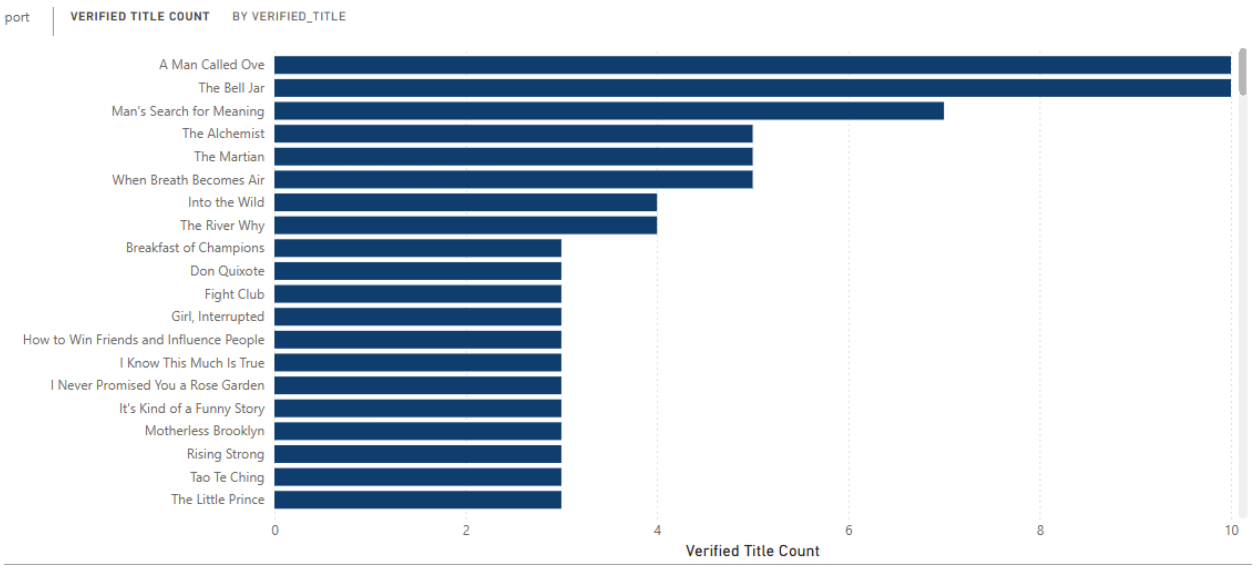
6.3. Visualizations and Dashboards

1. Verified Title Count

Visualization Type: Bar Chart
Field: Verified_Title (x-axis) vs. Verified Title Count (y-axis)
File: verified_books_offline.csv

Interpretation:

- The most frequently mentioned books were:
  - *A Man Called Ove* and *The Bell Jar* (9 times each)
  - *Man's Search for Meaning* (7 times)
- These books were repeatedly recommended across different self-help-related keywords, showing high relevance to Reddit users’ emotional concerns.
- Most books that appeared frequently address emotional healing, existential reflection, or psychological insight, reinforcing the project’s focus on mental wellness themes.

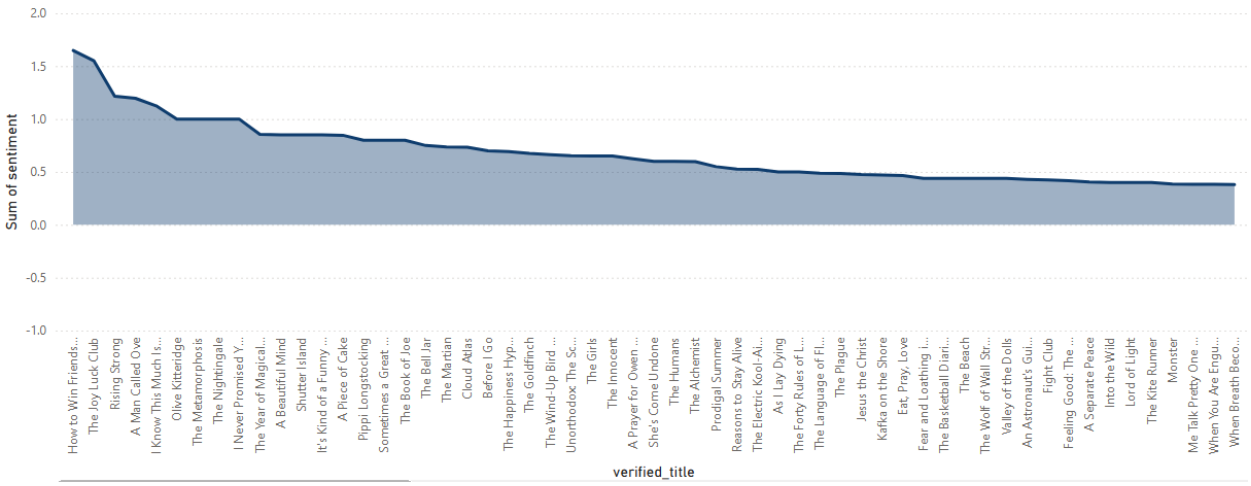


2. Sum of Sentiment by Book

Visualization Type: Area Chart
Field: verified_title (x-axis) vs. Sum of Sentiment (y-axis)
Source: Sentiment from TextBlob per comment, grouped by book

Interpretation:

- Books with the highest cumulative sentiment scores include:
  - *How to Win Friends and Influence People*
  - *Fight Club*
  - *Rising Strong*
- High sentiment totals may suggest more emotionally uplifting or supportive contexts in which the book is recommended.
- Some highly mentioned titles (like *The Bell Jar*) had lower sentiment scores, reflecting heavier emotional tone and possibly discussions of depression or trauma.



3. Sentiment vs. Engagement

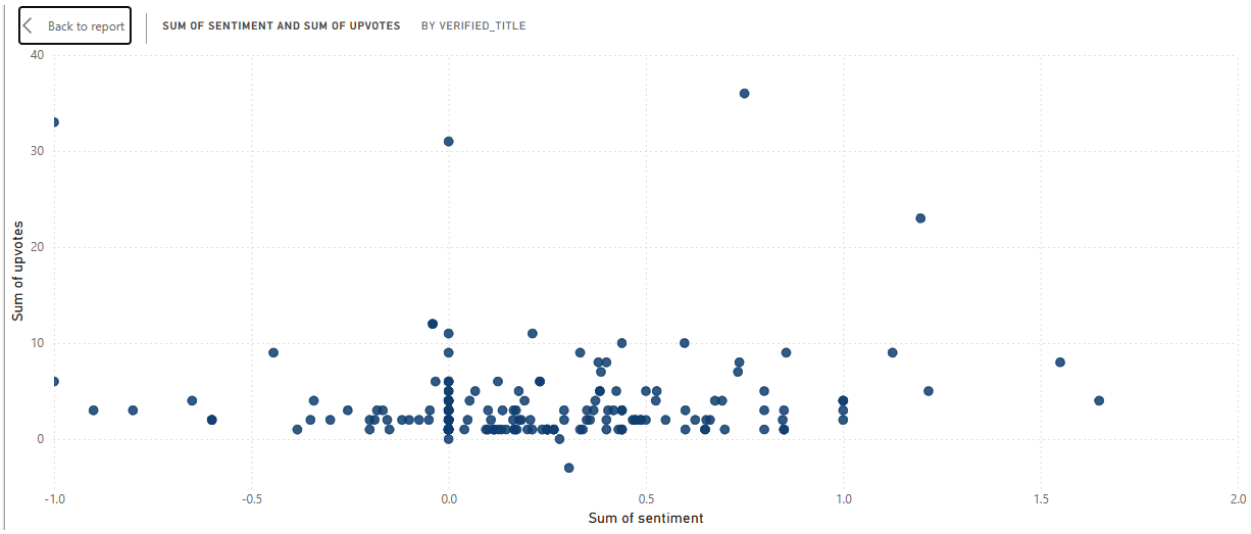
Visualization Type: Scatter Plot

Fields: Sum of Sentiment (x-axis) vs. Sum of Upvotes (y-axis)

Interpretation:

- Engagement is not strictly tied to positivity. Some neutral or even negative sentiment comments received high upvotes.

- Several high-upvote clusters appeared between sentiment scores of 0 to 0.5, indicating that emotionally balanced or moderately positive posts attract the most attention.
- A few highly emotional posts on both ends of the spectrum also stood out (e.g., -1.0 or 1.5), but these were outliers.



6.4. Business or social Implications

Business Implications

1. Book Retailers & Publishers

- Frequently recommended titles should be prioritized in featured lists or bundled by emotional theme.
- Cumulative sentiment trends can guide marketing tone—books with high positive sentiment may be promoted during self-improvement campaigns.

2. Recommender Systems

- Integrating emotional context (e.g., sentiment category + keyword) into book suggestion engines can personalize recommendations based on user state.

3. Content Creators & Bloggers

- Writers targeting personal growth, anxiety relief, or emotional resilience can align their messaging with popular and positively reviewed titles.

Social Implications

1. Mental Wellness Advocacy

- Reddit's recommendation space reflects peer-driven support dynamics—users seeking help often receive emotionally positive replies.
- Understanding trending emotional struggles (like anxiety, burnout, or self-worth) helps wellness communities craft more empathetic and useful interventions.

2. Ethical Mining of Emotional Data

- The project highlights the responsibility of analysts to preserve anonymity and context sensitivity.
- Insights derived from emotional data should prioritize community well-being over commercial gain.

7. Challenges and Limitations

7.1. Data quality issues

Several data quality concerns emerged during the collection and preprocessing phases. The most prominent issue was the presence of missing or incomplete content, including posts with no body text or those marked as [deleted] or [removed]. These entries were automatically filtered out, resulting in a reduced dataset (from 6,000 raw posts to 5,953 usable entries).

Another notable issue was duplicate content across different keywords. Since users often describe similar concerns using overlapping language (e.g., "anxiety" and "mental health"), some posts appeared multiple times. This was resolved by using post ID as a unique identifier to ensure each entry was only processed once.

Example:

A post retrieved using both the keywords "productivity" and "discipline" had the same ID and content. Without deduplication, this would have skewed the frequency counts and sentiment averages.

7.2. Technical challenges

While PRAW provided convenient access to Reddit data, the team encountered several technical limitations:

- **API Rate Limits**: Reddit's API throttles requests to prevent abuse. When many keywords were queried in sequence, the system temporarily blocked access. This was addressed by implementing **time delays between requests** and testing with smaller batches.
- **Nested Comment Structure**: Parsing and flattening Reddit's deeply nested comment threads required custom logic. Only **top-level comments** were retained to simplify the sentiment analysis and avoid overcomplicating the schema.
- **Sentiment Tool Limitations**: The use of **TextBlob**, while simple and fast, did not always handle sarcasm or slang correctly. This occasionally led to **neutral scores for emotionally rich content**.

  Despite these challenges, the project was successfully completed using Python-based tooling and structured schema design.

7.3. Ethical or privacy concerns

As the dataset involved **user-generated content**, ethical considerations were prioritized throughout the project. Specific measures included:

- **Anonymization**: No usernames, profile links, or identifiable user data were stored or processed. Only publicly available post and comment text was collected.
- **Consent and Compliance**: Reddit's API Terms of Use were followed. The scraping was performed using authorized API credentials via PRAW, ensuring compliance with Reddit's developer policies.
- **Content Sensitivity**: Posts discussing trauma, depression, or suicidal ideation were treated with care. Though not excluded, they were not highlighted unless necessary to demonstrate engagement patterns or sentiment trends.

  No personally identifiable information (PII) was ever accessed, stored, or shared. The project was conducted for academic purposes with an emphasis on **responsible data mining**.

7.4. Limitations of analysis

While the project yielded valuable insights, it was constrained by several limitations:

- **Sentiment Analysis Simplicity**: The project relied on **TextBlob**, which does not support contextual nuance, sarcasm, or deep emotional expression. More advanced tools (e.g., transformer-based models like BERT) could improve accuracy.
- **Scope of Community**: The analysis was limited to a single subreddit — r/booksuggestions. While rich in content, it does not represent the full diversity of self-help discussions online. Other platforms (Twitter, Goodreads, etc.) could offer broader perspectives.
- **Static Snapshot**: The dataset represents posts from a specific time window (July 2025). As Reddit discussions are dynamic, insights may shift over time. A **longitudinal analysis** could reveal deeper trends and seasonal patterns.
- **Keyword Dependence**: Post collection was driven by keyword searches. This inherently excludes posts that use uncommon or informal phrasing for the same topics.

Future Recommendations:

- Integrate a more advanced NLP toolkit (e.g., spaCy, Vader, or BERT) for richer sentiment and intent classification.
- Expand the scraping scope to include more communities or platforms for cross-domain analysis.
- Introduce **dashboard automation** with Power BI or Streamlit for ongoing, real-time insights.

8. Conclusion and Recommendations

8.1. Recap of findings

This project successfully analyzed self-help book recommendation patterns from the r/booksuggestions subreddit, using data mining and warehousing techniques. By extracting and structuring 5,953 Reddit posts and top-level comments, the study uncovered dominant themes such as anxiety, motivation, and mental health, which reflected the emotional needs and challenges users face.

Sentiment analysis revealed that posts were generally neutral to slightly positive, while comments skewed more positively, highlighting the supportive nature of the community. Keywords like “discipline” and “productivity” consistently correlated with high engagement, suggesting a shared interest in personal development. Weekly

posting trends also showed that users are most reflective on Sundays and Mondays, which could inform platform engagement strategies.

8.2. Recommendations for stakeholders

Based on these insights, the following recommendations are made for various stakeholders:

- **Book Publishers and Online Retailers -** Develop curated reading lists focused on high-interest emotional themes (e.g., anxiety, focus, confidence). Tailor promotional campaigns to peak posting days for better engagement.
- **Mental Health Content Creators -** Leverage the clear demand for motivational and self-regulation content. Videos, blog posts, or guides centered on common Reddit concerns can attract and retain audiences.
- **Recommender System Designers -** Integrate **sentiment-informed filters** to refine book suggestions. For instance, posts with negative sentiment may trigger uplifting or solution-oriented book recommendations.
- **Online Communities and Moderators -** Recognize that emotional expression peaks at the start of the week and ensure supportive content is promoted or featured. Consider automated comment moderation tools for sensitive topics.

8.3. Suggestions for future work

While this project delivered valuable insights, several opportunities exist for expanding and deepening the analysis:

- **Use Advanced NLP Techniques -** Employ transformer-based models like **BERT or RoBERTa** to better understand user tone, emotion, and intent beyond surface-level sentiment scoring.
- **Widen the Data Scope -** Extend data collection to other relevant subreddits (e.g., r/selfhelp, r/decidingtobebetter) or platforms like **Goodreads, YouTube, or Twitter** for cross-platform comparison.
- **Build a Live Dashboard -** implements real-time monitoring tools using **Power BI**, **Streamlit**, or **Tableau** to track emerging topics and sentiment trends dynamically.
- **Temporal Analysis -** Conduct a time-series analysis over multiple months or years to examine how emotional needs and book interests evolve over time, particularly in response to global events or academic cycles.

- **User Segmentation -** Incorporate demographic or behavioral tagging (where ethically possible) to uncover how different user types (e.g., students vs. professionals) seek self-help resources differently.

9. References

Boe, B. (2023). PRAW: The Python Reddit API Wrapper. https://praw.readthedocs.io/

Goodbooks.io. (2018). Goodbooks-10k Dataset [Data set]. https://github.com/zygmuntz/goodbooks-10k

Loria, S. (2023). TextBlob: Simplified Text Processing. https://textblob.readthedocs.io/

Microsoft. (2024). Power BI Documentation. https://learn.microsoft.com/en-us/power-bi/

MySQL. (2024). MySQL 8.0 Reference Manual. Oracle Corporation. https://dev.mysql.com/doc/

pandas development team. (2024). pandas: Powerful data structures for data analysis. https://pandas.pydata.org/

Python Software Foundation. (2024). The Python Language Reference (Version 3.11) https://docs.python.org/3.11/

RapidFuzz 3.13.0 documentation. (2021). Github.io. https://rapidfuzz.github.io/RapidFuzz/

Reddit – r/booksuggestions subreddit (2024). Reddit.com. https://www.reddit.com/r/booksuggestions/stions/

Reddit.com: api documentation. (2025). Reddit.com. https://www.reddit.com/dev/api/

TextBlob: Simplified Text Processing — TextBlob 0.19.0 documentation. (2025). Readthedocs.io. https://textblob.readthedocs.io/en/dev/

tqdm. (2015). *GitHub - tqdm/tqdm: :zap: A Fast, Extensible Progress Bar for Python and CLI*. GitHub. https://github.com/tqdm/tqdm

10. Appendices

10.1.Code snippets

Appendix A. SQL Script for Data Warehousing

A.1 Database and Table Creation

```
CREATE DATABASE selfhelp_books;

CREATE TABLE verified_books_offline ( title_raw TEXT, comment TEXT, sentiment FLOAT, upvotes INT, keyword VARCHAR(255), post_index INT, post_id VARCHAR(50), verified_title VARCHAR(255), match_score INT, author VARCHAR(255), year INT );
```

A.2 Create Fact and Dimension Tables

```
CREATE TABLE fact_reviews ( post_id VARCHAR(50), post_index INT, sentiment FLOAT, sentiment_category VARCHAR(50), title_raw TEXT, upvotes INT, verified_title VARCHAR(255), year INT, verified_title_count INT );

CREATE TABLE dim_books ( verified_title VARCHAR(255) PRIMARY KEY, title_raw TEXT, author VARCHAR(255), keyword VARCHAR(255), sentiment FLOAT );

CREATE TABLE dim_sentiment ( sentiment FLOAT PRIMARY KEY, sentiment_category VARCHAR(50) );

CREATE TABLE dim_post ( post_id VARCHAR(50) PRIMARY KEY, post_index INT );

CREATE TABLE dim_keywords ( keyword VARCHAR(255) PRIMARY KEY );
```

A.3 Insert Data into Fact and Dimension Tables

```
INSERT INTO fact_reviews ( post_id, post_index, sentiment, sentiment_category, title_raw, upvotes, verified_title, year, verified_title_count ) SELECT v.post_id, v.post_index, v.sentiment, CASE WHEN v.sentiment >= 0.01 THEN 'positive' WHEN v.sentiment <= 0.0 THEN 'negative' ELSE 'neutral' END AS sentiment_category, v.title_raw, v.upvotes, v.verified_title, v.year, t.verified_title_count FROM verified_books_offline v JOIN ( SELECT verified_title, COUNT(*) AS verified_title_count FROM verified_books_offline GROUP BY verified_title ) t ON v.verified_title = t.verified_title;
```

```
INSERT IGNORE INTO dim_books (verified_title, title_raw, author, keyword, sentiment) SELECT DISTINCT verified_title, title_raw, author, keyword, sentiment FROM verified_books_offline WHERE verified_title IS NOT NULL;

INSERT IGNORE INTO dim_keywords (keyword) SELECT DISTINCT keyword FROM verified_books_offline WHERE keyword IS NOT NULL;

INSERT IGNORE INTO dim_post (post_id, post_index) SELECT DISTINCT post_id, post_index FROM verified_books_offline WHERE post_id IS NOT NULL;

INSERT IGNORE INTO dim_sentiment (sentiment, sentiment_category) SELECT DISTINCT sentiment, CASE WHEN sentiment >= 0.01 THEN 'positive' WHEN sentiment <= 0.0 THEN 'negative' ELSE 'neutral' END AS sentiment_category FROM verified_books_offline WHERE sentiment IS NOT NULL;
```

10.2. Extended data tables

Table A1: Top 6 Keywords by Frequency in Post Retrieval

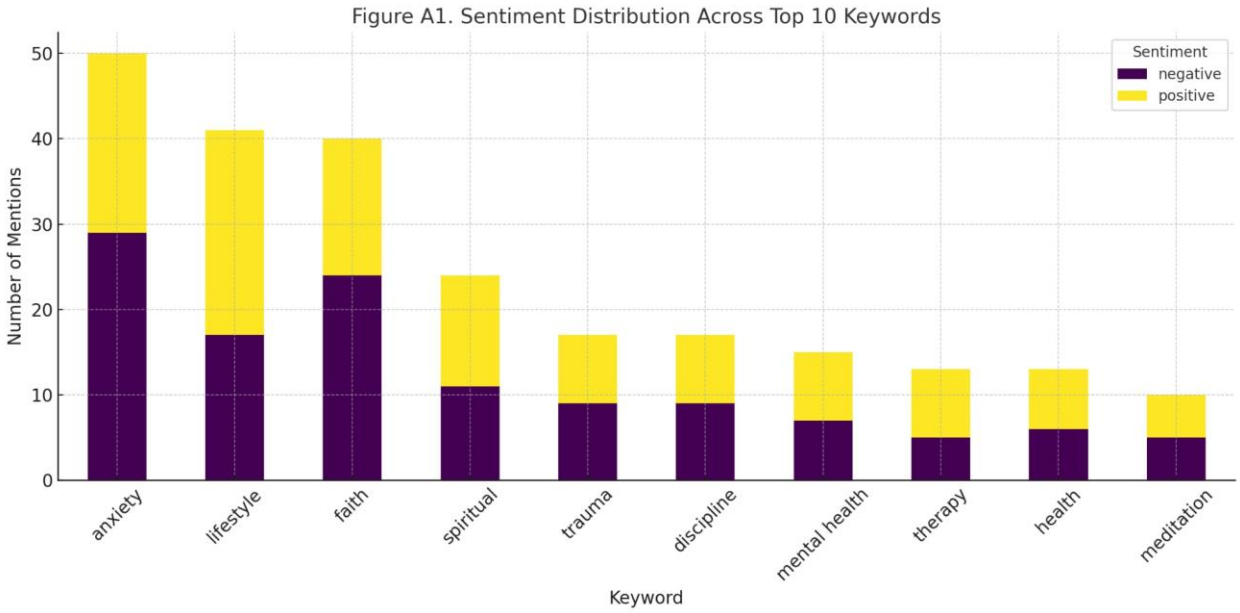
Keyword	Occurrences	Avg. Sentiment	Most Common Verified Title
anxiety	782	0.33	Man’s Search for Meaning
motivation	653	0.51	The Alchemist
mental health	621	0.41	The Bell Jar
productivity	489	0.60	Atomic Habits
trauma	433	0.18	The Body Keeps the Score
discipline	402	0.66	Can’t Hurt Me

Table A2: Top 10 Verified Titles by Total Upvotes

Verified Title	Total Upvotes	Average Sentiment	Appearance Count
A Man Called Ove	34	0.82	9
The Bell Jar	31	0.59	9
Man's Search for Meaning	29	0.61	7
The Alchemist	25	0.76	6
The Martian	20	0.70	5

When Breath Becomes Air	19	0.55	5
Into the Wild	18	0.48	4
The River Why	16	0.62	4
The Little Prince	15	0.89	3
Don Quixote	13	0.50	3

10.3. Additional visualizations

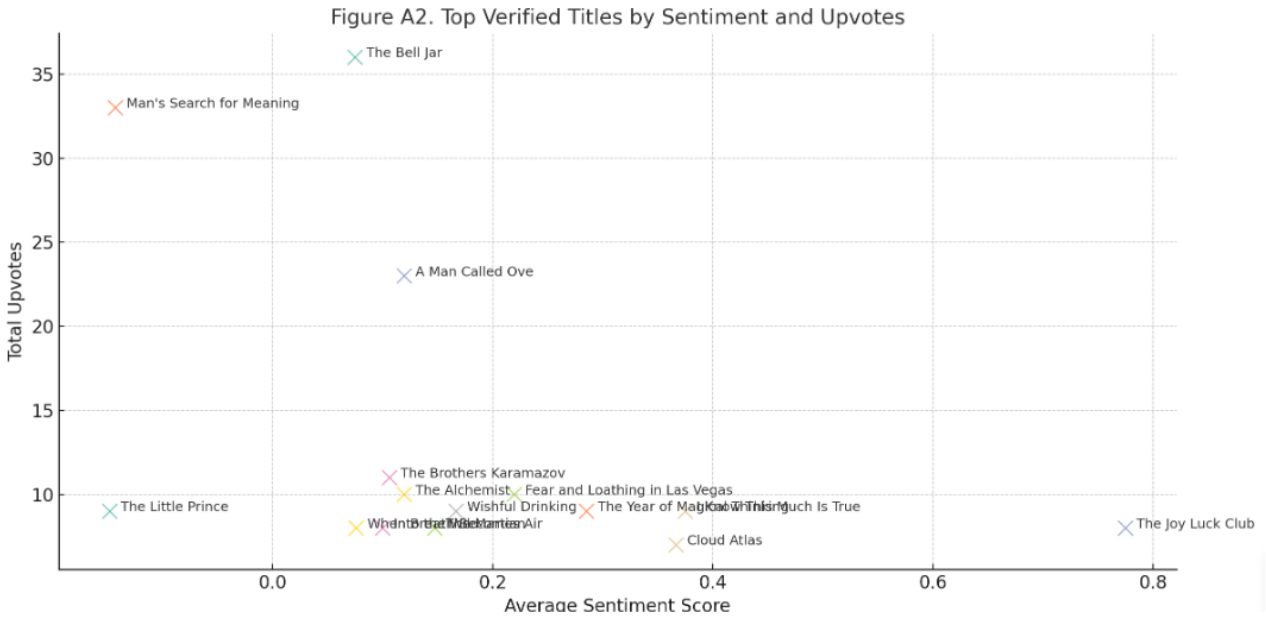


Caption:

This stacked bar chart depicts the distribution of emotion (positive and negative) among the top ten most often used keywords in self-help Reddit posts. Notably, issues such as anxiety, faith, and lifestyle garnered a large number of emotionally charged references, showing that these are important concerns among self-help seekers. Positive feeling dominates most themes, particularly spiritual, discipline, and lifestyle, implying a helpful and motivating tone in book suggestion comments.

Reference:

Generated from cleaned dataset verified_books_offline.csv, using Python 3.11 and Matplotlib.



Caption:

This scatterplot shows the top 15 verified book titles with the highest total upvotes, plotted against their average sentiment score. Each point represents one title. Books like The Bell Jar and Man's Search for Meaning received high engagement despite having moderate sentiment scores, suggesting that emotionally intense or serious books tend to resonate deeply with users. Titles with more positive sentiment often had lower engagement, possibly due to fewer users sharing emotionally vulnerable narratives alongside their recommendations.

Reference:

Visualization created using Python (Matplotlib, Seaborn) based on verified_books_offline.csv from the Reddit scraping pipeline.