Sentiment-Enhanced Drug Recommendation System

Harshit Bhushan, 210420, harshitb21@iitk.ac.in
Aman Sarwa, 210112, amansarwa21@iitk.ac.in
Ambuj Pyne, 210118, ambujp21@iitk.ac.in
Rahul Kumar, 210804, kumarrahul21@iitk.ac.in
Yash Sinha, 211195, yashs21@iitk.ac.in

CS685: Data Mining

Abstract

This project undertakes an analysis of drug reviews to classify sentiment and identify drug usage patterns related to specific health conditions. The dataset, sourced from Drugs.com, includes over 200,000 patient reviews on various drugs along with a 10 star patient rating reflecting overall patient satisfaction about the drug. Exploratory Data Analysis was conducted to uncover insights into common conditions, drug distributions, and prevalent sentiments. We employed two machine learning methods, LightGBM and deep learning Neural Network, along with an emotional dictionary to perform sentiment analysis on the dataset, where ratings of 5 and above are labeled as positive and below 5 as negative. Results indicate the effectiveness of both models, with the deep learning model achieving 71% and LightGBM achieving an accuracy of 90%. The combined findings of these models along with the sentiment score of the dictionary was then used make the best drug recommendation based on a specific condition. These findings demonstrate the feasibility of automated sentiment analysis in healthcare, offering potential applications for pharmaceutical companies and healthcare providers seeking to understand patient feedback.

1 Motivation of the Problem

Sentiment analysis of user-generated data in healthcare, such as drug reviews, provides valuable insights into patient experiences. Understanding these sentiments can:

- Help healthcare professionals identify the most effective drugs for specific conditions.
- Provide pharmaceutical companies with direct feedback from patients.
- Empower patients with information for better decision-making regarding their treatment options.

Despite the abundance of online reviews, deriving structured insights from this unstructured data remains a challenge, necessitating the use of advanced NLP and machine learning models.

2 Data Used

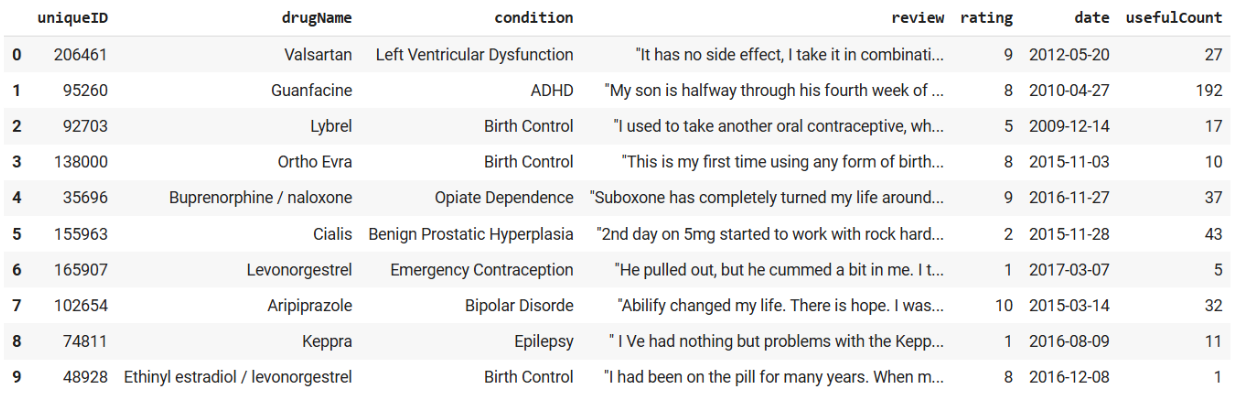
The dataset utilized in this analysis comprises Train and Test data sourced from the UCI Machine Learning Repository - Drug Review Dataset- Drug Review Dataset and includes crawled reviews from online pharmaceutical review sites. The sizes of the datasets are as follows:

- Training Set: (161297, 7)
- Test Set: (53766, 7)
- Combined Dataset: (215,063,7)

Dataset Structure:

The structure of the dataset is such that a patient with a unique ID purchases a drug to treat their condition and writes a review along with a rating for the purchased drug. The usefulCount variable is incremented if other users find the review helpful. The dataset contains seven key variables:

- **uniqueID**: Unique identifier for each review.
- **drugName**: Name of the drug.
- **condition**: Medical condition treated by the drug.
- **review**: Textual review of the drug by users.
- **rating**: User rating of the drug (1–10 scale).
- **usefulCount**: Number of users who found the review helpful.
- **date**: Date when the review was posted.

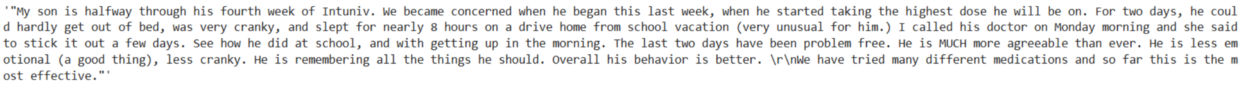


	uniqueID	drugName	condition	review	rating	date	usefulCount
0	206461	Valsartan	Left Ventricular Dysfunction	"It has no side effect, I take it in combinati...	9	2012-05-20	27
1	95260	Guanfacine	ADHD	"My son is halfway through his fourth week of ...	8	2010-04-27	192
2	92703	Lybrel	Birth Control	"I used to take another oral contraceptive, wh...	5	2009-12-14	17
3	138000	Ortho Evra	Birth Control	"This is my first time using any form of birth...	8	2015-11-03	10
4	35696	Buprenorphine / naloxone	Opiate Dependence	"Suboxone has completely turned my life around...	9	2016-11-27	37
5	155963	Cialis	Benign Prostatic Hyperplasia	"2nd day on 5mg started to work with rock hard...	2	2015-11-28	43
6	165907	Levonorgestrel	Emergency Contraception	"He pulled out, but he cummed a bit in me. I t...	1	2017-03-07	5
7	102654	Aripiprazole	Bipolar Disorde	"Abilify changed my life. There is hope. I was...	10	2015-03-14	32
8	74811	Keppra	Epilepsy	" I Ve had nothing but problems with the Kepp...	1	2016-08-09	11
9	48928	Ethinyl estradiol / levonorgestrel	Birth Control	"I had been on the pill for many years. When m...	8	2016-12-08	1

Figure 1: Dataset

Sample Review:

This is what a typical review looks like:



```
'"My son is halfway through his fourth week of Intuniv. We became concerned when he began this last week, when he started taking the highest dose he will be on. For two days, he coul
d hardly get out of bed, was very cranky, and slept for nearly 8 hours on a drive home from school vacation (very unusual for him.) I called his doctor on Monday morning and she said
to stick it out a few days. See how he did at school, and with getting up in the morning. The last two days have been problem free. He is MUCH more agreeable than ever. He is less em
otional (a good thing), less cranky. He is remembering all the things he should. Overall his behavior is better. \r\nWe have tried many different medications and so far this is the m
ost effective."'
```

Figure 2: Sample Review

3 Methodology

3.1 Exploratory Data Analysis (EDA)

- **Initial Observations**
  - **Unique ID Analysis:** We checked whether a single customer had written multiple reviews by comparing the unique number of uniqueID values with the length of the Train data. The results confirmed that no customer wrote more than one review, ensuring the uniqueness of entries.
  - **Relationship Between drugName and condition:**
    - The number of unique values for drugName and condition are 3671 and 917, respectively.
    - On average, there are approximately 4 drugs available for each condition.
    - Further analysis revealed that the top eight conditions have around 100 drugs each, which aligns with expectations.

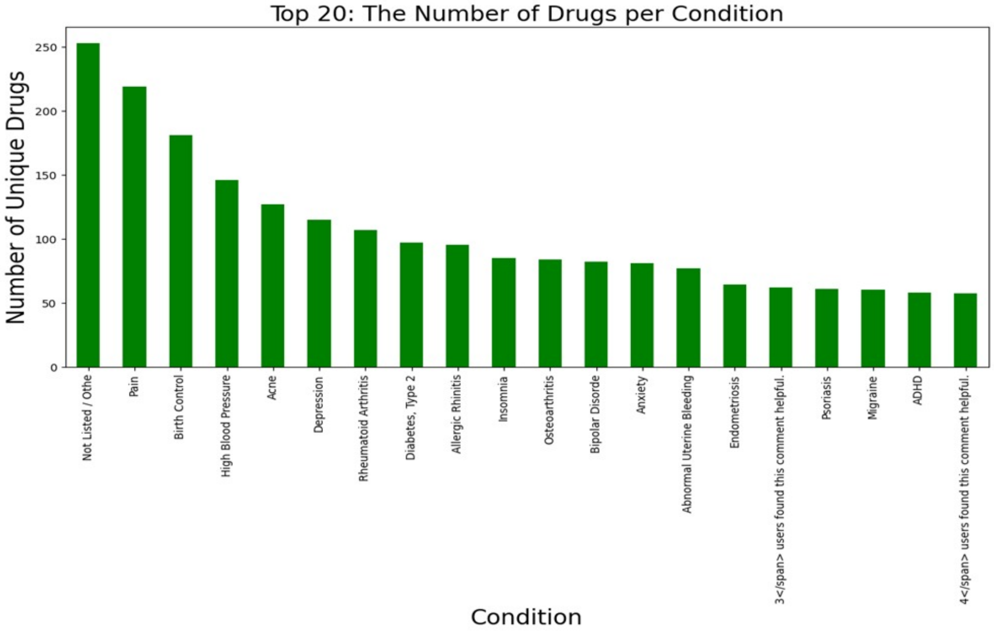


Figure 3

  - **Lowest Drug-to-Condition Ratio:** The lowest 20 conditions with respect to the number of drugs per condition were analyzed. All of these conditions had only one drug. Since recommendation systems require a minimum of two options to provide meaningful suggestions, conditions with fewer than two drugs will be excluded from the analysis.

- **Review Text Observations**
  - **HTML Artifacts:** Strings such as \r \n were observed in the text.
  - **Emotive Phrases:** Parenthetical expressions (e.g., ”(very unusual for him)”, ”(a good thing)”).

  - **Capitalized Words:** Words in uppercase (e.g., ”MUCH”).
  - **Encoding Errors:** Words like didn&#039;t instead of ”didn’t”.
  - **Special Characters:** Characters such as ... appeared frequently.

  All these inconsistencies were handled during preprocessing to clean the text data.

- **Word Cloud**

  To gain an initial understanding of the text data, several Word Cloud visualizations were generated to provide insights into common themes and keywords in the dataset.

  - Word cloud for the reviews with some custom stopwords.
  - Word cloud for the Postive sentiments.
  - Word cloud for the Negative sentiments.
  - Word cloud of cleaned reviews.

- **N-gram Classification**

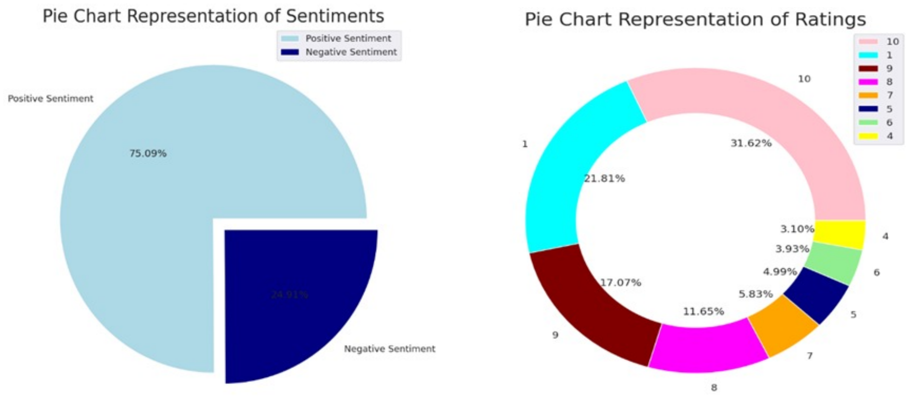
  The classification was analyzed using n-grams from 1-gram to 4-gram to determine the most effective corpus for emotion detection.

  - **1-Gram:** The top five words for both negative and positive sentiments were nearly identical in content, differing only in order. This indicated poor classification performance.
  - **2-Gram:** Although 2-grams performed slightly better than 1-grams, issues persisted, such as treating ”side effects” and ”side effects.” as different entities, necessitating preprocessing. Terms like ”side effects,” ”weight gain,” and ”highly recommend” appeared frequently.
  - **3-Gram:** A noticeable improvement in distinguishing positive and negative sentiments was observed. Examples like ”bad side effects,” ”birth control pills,” and ”negative side effects” highlighted the difference. However, context reversal due to missing negations (e.g., ”not”) remained a concern.
  - **4-Gram:** The 4-gram analysis demonstrated significantly better sentiment classification. It resolved many issues observed in lower n-grams, making it the optimal choice for building a deep learning model.

- **Missing Values** The dataset contained missing values in certain fields, with the overall missing value percentage calculated as **0.5579%**. Since this percentage is below the threshold of **1%**, all entries with missing values were removed from the dataset to ensure data integrity and consistency in the analysis.

3.2 Data Preprocessing

- **Handling Missing Values**: The “Condition” column had a few missing values (0.56% of the total data), which were removed to maintain data integrity and data with less than one unique drug per condition was also excluded.



(a) Sentiment representation

(b) Ratings representation

Figure 4

- **Filtering Conditions with Insufficient Data**: Conditions associated with only one unique drug were identified and removed from the dataset to ensure that sufficient variability exists within each condition for meaningful analysis. After removing these entries, the dataset was re-indexed to maintain continuity.
- **Cleaning Reviews**: To prepare the review text data for analysis, a robust cleaning pipeline was implemented. This pipeline involves the following steps:
  1. *HTML Tag Removal*: HTML tags present in the review text were removed using the `BeautifulSoup` library to extract clean text.
  2. *Lowercasing*: All text was converted to lowercase to standardize the format and ensure case-insensitive processing.
  3. *Special Character Removal*: Non-letter characters, non-ASCII characters, and excess whitespace were removed using regular expressions.
  4. *Stopword Removal*: Common stopwords were filtered out to retain only meaningful words, using the NLTK `stopwords` corpus.
  5. *Stemming*: Words were reduced to their root forms using the Snowball Stemmer for consistency in linguistic analysis.
  6. *Feature Integration*: The cleaned and processed reviews were added as a new column, `review_clean`, to the dataset for subsequent analysis.

3.3 Machine Learning Models Used

- **Deep Learning Model using N-Grams:** It is a sequential Deep Neural Network (DNN).

  Architecture:

  – **Input layer:** Top 20,000 features (4-grams) from the CountVectorizer.

  – **Hidden layers:** Multiple fully connected layers with ReLU activations.

  – **Regularization:** Batch normalization and dropout to prevent overfitting.

  - **Optimizer:** Adam with a binary cross-entropy loss function
  - **Output layer:** 1 neuron with a sigmoid activation function for binary classification.

- **LightGBM Model 1:** A LightGBM classifier to predict the sentiment of reviews using the feature **usefulCount**.

  Hyperparameters:

  - **n_estimators**: 10000
  - **learning_rate**: 0.05
  - **num_leaves**: 30
  - **max_depth**: 7

- **LightGBM Model 2:** A hyperparamter-tuned LightGBM classifier trained on dataset enriched with text preprocessing, sentiment analysis, and feature engineering.

  Feature Engineering:

  - Used TextBlob to compute sentiment polarity scores ranging from -1 (negative) to +1 (positive).
  - Extracted features such as sentence count, word count, unique word count, letter count, punctuation frequency, uppercase word frequency, and title-case word frequency, no. of stopwords and average word length.
  - Used Label Encoder for the drugName and condition columns, converting each unique value into a unique numeric label, enabling machine learning model to process the data efficiently.

  Hyperparameters:

  - **n_estimators**: 10000
  - **learning_rate**: 0.1
  - **num_leaves**: 30
  - **max_depth**: 10

3.4 Sentiment Analysis Using Emotional Dictionary

In this step, sentiment analysis was performed using an emotional dictionary, which categorizes words into positive and negative sentiments. The sentiment of each review was determined based on the ratio of positive and negative words.

1. **Dictionary Import**: A Harvard Emotion Dictionary was imported, which contains a list of words categorized as either ”Positive” or ”Negative”. This dictionary was used to assess the sentiment of each review.

2. **Positive Word List Construction**: A list of positive words was created by filtering out words marked as ”Positive” in the dictionary. Each word was preprocessed to remove numbers, special characters, and standardized to lowercase to ensure consistency.

3. **Negative Word List Construction**: Similarly, a list of negative words was created by filtering out words marked as ”Negative” in the dictionary, following the same preprocessing steps as for the positive words.

4. **Positive Word Count**: Using the `CountVectorizer` from `sklearn`, the number of positive words in each review was counted based on the predefined positive word list. This count was stored in a new column, `num_Positiv_word`.

5. **Negative Word Count**: The same process was applied for negative words, using the negative word list, and the result was stored in a new column, `num_Negativ_word`.

6. **Sentiment Calculation**: The sentiment of each review was determined by calculating the ratio of positive words to the total number of positive and negative words. A ratio of 0.5 or higher was classified as positive sentiment (1), while a ratio below 0.5 was classified as negative sentiment (0). If the ratio was exactly 0.5, the sentiment was marked as neutral (0.5). This classification was stored in a new column, `sentiment_by_dic`.

This approach allowed for the automatic classification of sentiment based on the emotional tone conveyed by the words in the reviews, providing a quantitative measure of sentiment.

3.5 Drug Recommendation

The drug recommendation system was designed to suggest the most appropriate drug for a given condition based on multiple factors, including deep learning predictions, LightGBM predictions, sentiment analysis by dictionary, multiplied with the normalised useful count.

1. **Prediction Integration**: First, predictions from different models were combined. The deep learning-based predictions, LightGBM-based predictions, and Sentiment Dictionary-based predictions were summed up and weighted by the number of useful reviews. This resulted in a new column, `total_pred`, representing the overall recommendation score for each drug-condition pair.

2. **Data Aggregation**: The data was then grouped by condition and drug name, with the mean of the `total_pred` values computed for each group. This aggregation process allowed for a summary of how each drug performed across different conditions.

3. **Sorting by Prediction Score**: The resulting dataframe was sorted in descending order by the `total_pred` mean value, so that the drugs with the highest predicted recommendation score appeared at the top.

4. **Drug Recommendation Function**: A function was created to recommend the drug with the highest predicted score for a given condition. The function takes a condition as input, looks up the corresponding drugs, and returns the drug with the highest predicted value. If no drugs are found for the given condition, the function prints a message indicating this.

5. **User Input**: The system allows users to input a condition, and it will output the drug with the highest recommendation score for that condition.

4 Results

4.1 Model Performance

- **Deep Learning Model**: Achieved 71% accuracy and F1 score of 0.65 with slight bias towards positive sentiment due to data imbalance.

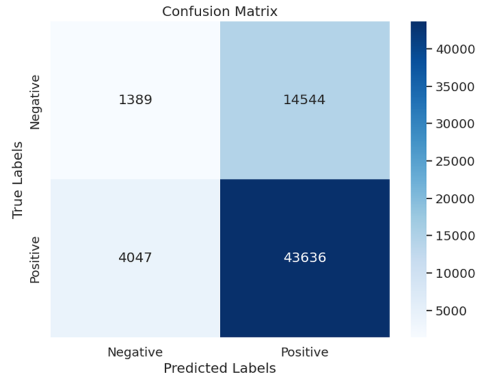


Figure 5: Confusion matrix of Deep Learning

- **LightGBM Model 1**: Achieved 75% accuracy and F1 score of 0.86, with useful count and sentiment scores being the features.

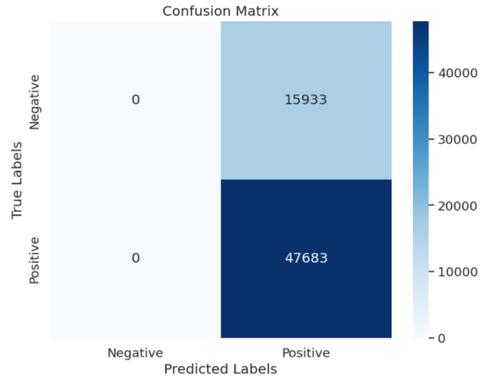


Figure 6: Confusion matrix of LGBM 1

- **LightGBM Model 2**: Achieved 91% accuracy and F1 score of 0.91 with hyperparameter tuning.

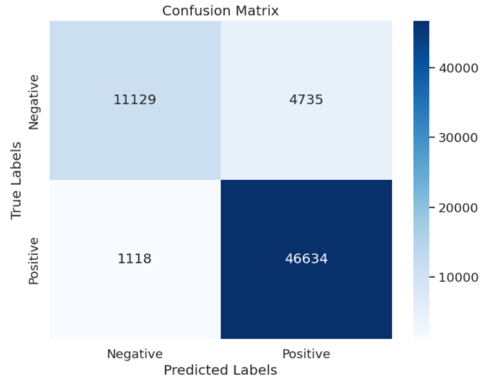
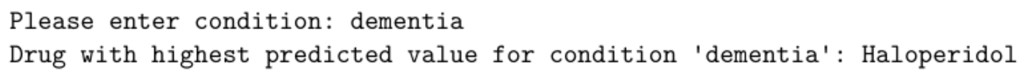


Figure 7: Confusion matrix of LGBM 2

- **Drug Recommendation System:** The system allows users to input a condition, and it will output the drug with the highest recommendation score for that condition.



```
Please enter condition: dementia
Drug with highest predicted value for condition 'dementia': Haloperidol
```

Figure 8: Drug Recommendation

5 Conclusions and Future Directions

5.1 Conclusions

This project demonstrates the feasibility and utility of sentiment analysis in understanding patient feedback on drugs for various conditions. By integrating advanced machine learning models such as LightGBM and a deep neural network, the project achieved high accuracy in sentiment classification and laid the groundwork for automated drug recommendation systems. These findings underscore the potential to enhance decision-making for patients, healthcare providers, and pharmaceutical companies.

Despite these advancements, challenges remain in addressing nuanced sentiments, especially for negative reviews, and improving the interpretability of model outputs. Nonetheless, the models showcased in this study provide a strong starting point for leveraging patient reviews in healthcare analytics.

5.2 Future Directions

- **Model Enhancement:** Incorporate advanced transformer-based architectures like BERT or RoBERTa for more nuanced sentiment classification and improved handling of imbalanced datasets. Explore ensemble techniques combining the strengths of traditional ML and neural network models.
- **Attribute-Specific Analysis:** Expand the scope to analyze sentiment around specific drug attributes, such as efficacy, side effects, and pricing, to offer more granular insights.
- **Real-Time Applications:** Develop an interactive dashboard that performs real-time sentiment analysis of drug reviews, enabling dynamic updates and user-friendly visualizations.
- **Broader Dataset Analysis:** Extend the analysis to larger, diverse datasets across different platforms to generalize findings and improve the robustness of the recommendation system.

6 Team Contributions

- **Harshit Bhushan (210420)**: Proposed the project idea, implemented the machine learning models, developed the drug recommendation system, and coordinated the overall project workflow.
- **Aman Sarwa (210112)**: Contributed to EDA, Developed the deep learning model and performed hyperparameter tuning to optimise the models.
- **Ambuj Pyne (210118)**: Conducted feature engineering and implemented sentiment analysis using TextBlob and emotional dictionary.
- **Rahul Kumar (210804)**: Contributed to EDA and data preprocessing; compiled the final report and coordinated team discussions.
- **Yash Sinha (211195)**: Contributed in EDA, deep learning model development; compiled the final report and coordinated team discussions.