

# Drug Sentiment Analysis

## Using Review Text and User Ratings

Group 7

Lecturer: PhD. Le Thi Khuyen

Le Thu Hang  
Mai Tra Giang  
Nguyen Thi Tuyet May  
Vo Thi Minh Phuong

May 8, 2025

# Table Contents

- 1 Dataset Overview
- 2 Exploratory Data Analysis
- 3 Data Pipeline
- 4 Data Preprocessing
- 5 Statistic Models
- 6 LLM Models
- 7 Results
- 8 Conclusion

# Dataset Overview

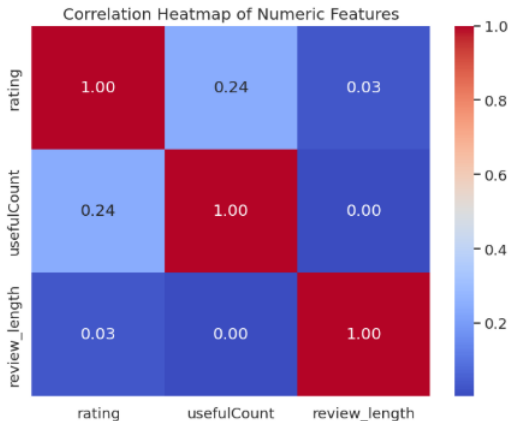
- The dataset contains 110,811 drug reviews from patients, including 4 object features, 4 numerical features.

	patient_id	drugName	condition	review	rating	date	usefulCount	review_length
0	89879	Cyclosporine	keratoconjunctivitis sicca	"i have used restasis for about a year now and...	2.0	April 20, 2013	69	147
1	143975	Etonogestrel	birth control	"my experience has been somewhat mixed. i have...	7.0	August 7, 2016	4	136
2	106473	Implanon	birth control	"this is my second implanon would not recommen...	1.0	May 11, 2016	6	140
3	184526	Hydroxyzine	anxiety	"i recommend taking as prescribed, and the bot...	10.0	March 19, 2012	124	104
4	91587	Dalfampridine	multiple sclerosis	"i have been on ampyra for 5 days and have bee...	9.0	August 1, 2010	101	74

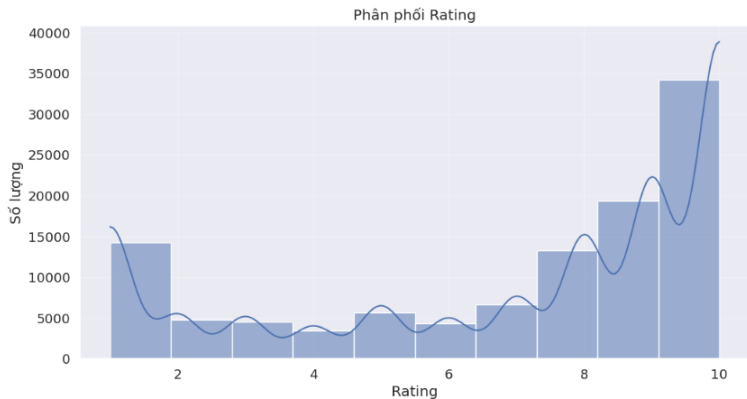
- Our problem is to predict positive or negative sentiment from review text by first converting ratings into binary sentiment labels base on drug dataset

# EDA - Heatmap correlation

The length of the review has the lowest correlation with all other features (0.03 with rating, 0.00 with usefulCount), so it should be dropped



## ● Rating Distribution



The rating distribution is highly imbalanced, with a significant concentration at the highest score (10), indicating a strong positive bias in user reviews.

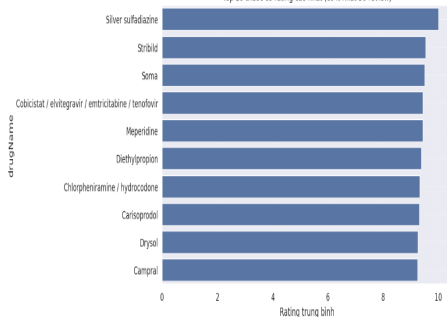
# EDA - Rating Score

- Related features with rating score

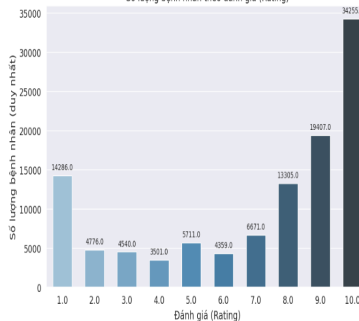
Xu hướng Rating trung bình theo thời gian



Top 10 thuốc có rating cao nhất (có ít nhất 30 review)

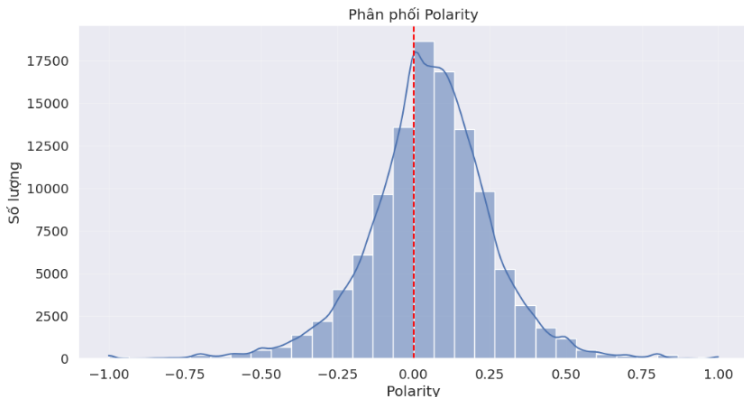


Số lượng bệnh nhân theo đánh giá (Rating)



# EDA - Polarity index

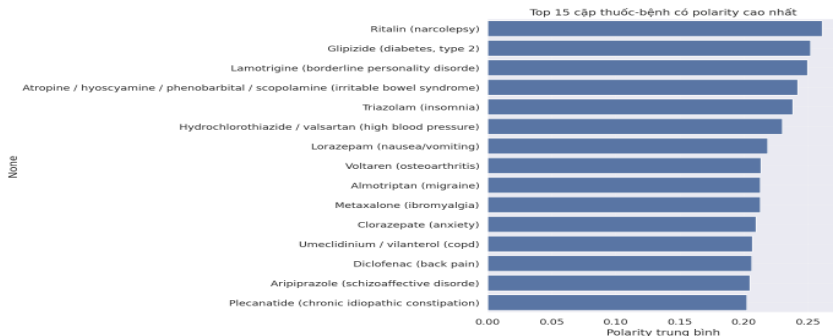
- Polarity index is a measure in sentiment analysis to determine how positive or negative a text is base on review text.



The bell-shaped chart shows that most polarity values cluster around 0, indicating neutral or mixed reviews. However, the distribution slightly leans positive, as the area on the right side of zero appears slightly larger.

# EDA - Polarity index

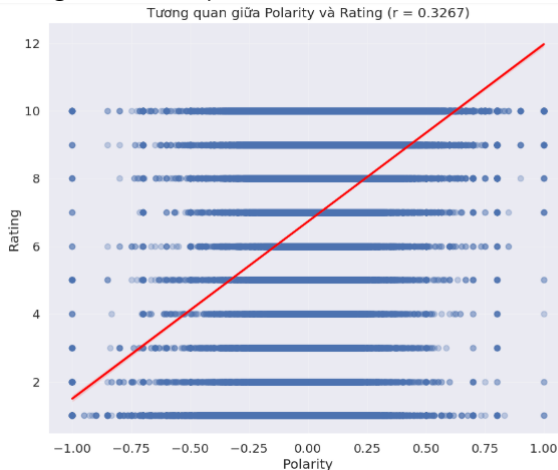
- Related feature with polarity index





# EDA - Rating and Polarity relationship

There is a weak positive correlation( $r = 0.3267$ )between polarity and rating,suggesting that more positive reviews tend to have higher ratings.



# Data Pipeline

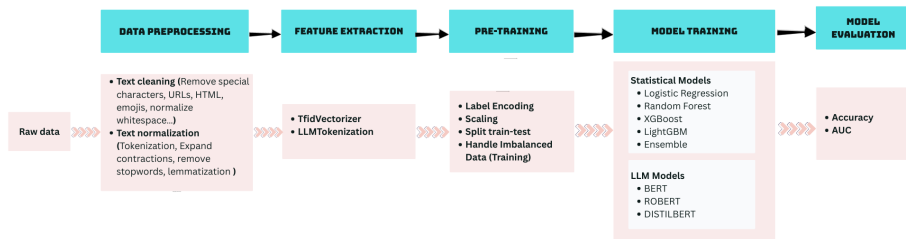


Figure: Overall NLP Model Pipeline

- Convert all text to lowercase to ensure consistency.
- Remove HTML tags, email addresses, URLs, emojis, and special characters.
- Correct common spelling errors and remove duplicate entries.
- Tokenize sentences into words using NLTK.
- Expand contractions (e.g., won't → will not), normalize whitespace.
- Remove stopwords (e.g., “the”, “is”, “in”) to reduce noise.
- Apply lemmatization to reduce words to their base form (e.g., “running” → “run”).
- Detect named entities, sarcasm, and perform synonym replacement.

# Stat-Model Setup

- Convert ratings into binary classes: negative ( $< 5$ ) and positive ( $> 5$ ).
- Apply TF-IDF vectorization to transform text into numerical features.
- Use this setup to build a basic text classification problem.
- Split the dataset into training and test sets before applying the pre-trained model.

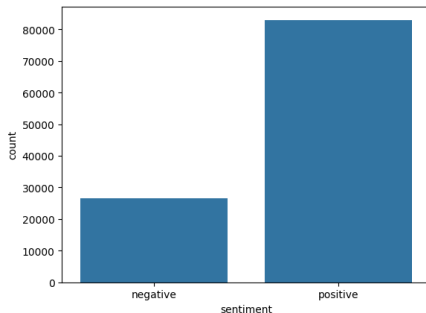


Figure: Sentiment Distribution

- Use Random OverSampling (ROS) to replicate minority class samples.
- Apply SMOTE (Synthetic Minority Over-sampling Technique) to generate synthetic samples.
- Resampling is performed only on the training set to avoid data leakage.

- Train multiple classification models on TF-IDF features:
  - Logistic Regression
  - Random Forest
  - XGBoost
  - LightGBM
- Evaluate individual model performance on the test set.
- Combine models using ensemble techniques:
  - Voting Classifier (hard/soft voting)
  - Stacking Classifier (meta-model approach)
- Select the best-performing model for deployment

- Use the `token_clean_contracted` column as input for the language model.
- Convert rating into binary labels: positive ( $> 5$ ) and negative ( $< 5$ ).
- Encode the text using a pre-trained transformer tokenizer (e.g., RoBERTa, BERT).
- Split the dataset into training and test sets before fine-tuning the model.

- **Multi-technique class balancing approach:**

- Weighted loss function calculation based on class distribution
- Three-pronged text augmentation for minority class: - WordNet synonym replacement - Back-translation (EN→FR→EN) - Easy Data Augmentation (random word swap)

- **Implementation details:**

- Equal distribution of augmentation techniques
- RoBERTa tokenization with consistent parameters
- Persistent storage of augmented data for reproducibility



- **Dataset preparation:**

- TensorDataset construction with encoded inputs and labels
- WeightedRandomSampler for additional sampling balance
- DataLoader configuration with appropriate batch sizes

- **Advanced loss functions:**

- Custom Focal Loss implementation ( $\gamma = 2.0$ )
- Class-weighted loss to penalize minority class errors

- **Model training pipeline:**

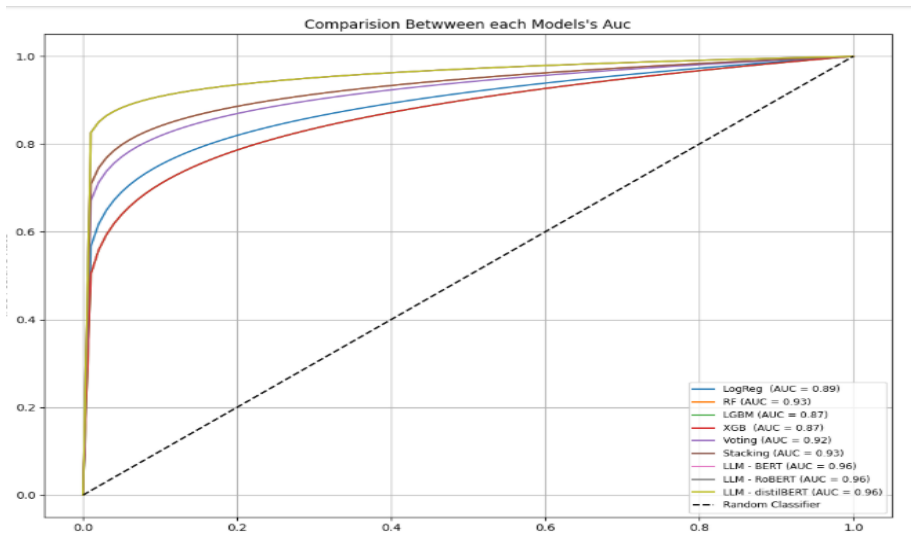
- Three pre-trained models: RoBERTa, BERT, DistilBERT
- AdamW optimizer with low learning rate ( $2e^{-5}$ )
- Consistent evaluation metrics across models

# Results

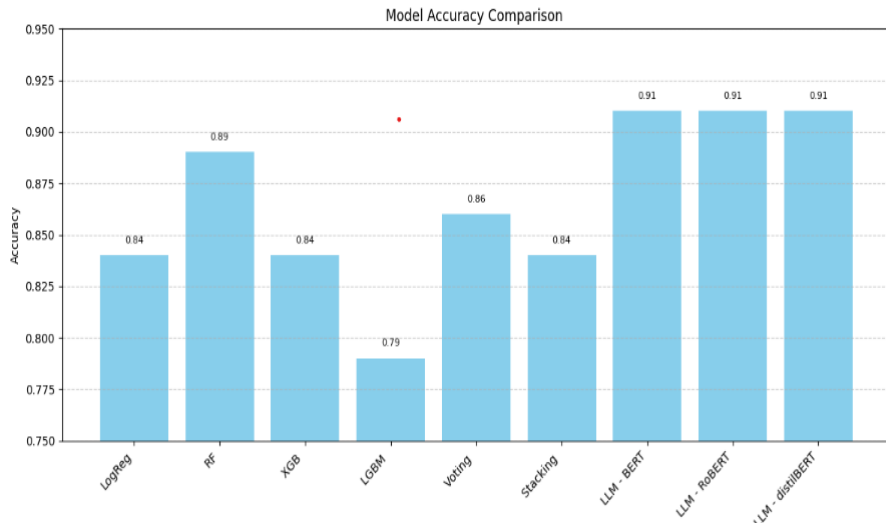
	Model	Accuracy	Auc
Statistical Model	Logistics Regression (ROS)	0.84	0.89
	Random Forest (ROS)	<b>0.89</b>	<b>0.93</b>
	LightGBM (ROS)	0.79	0.87
	XGBoost (ROS)	0.84	0.87
	Logistics Regression (SMOTE)	0.82	0.89
	Random Forest (SMOTE)	0.88	<b>0.93</b>
	LightGBM (SMOTE)	0.79	0.87
	XGBoost (SMOTE)	0.79	0.87
	Voting Classifier	0.86	0.92
	Stacking Classifier	0.84	<b>0.93</b>
LLM	LLM - BERT	0.91	0.96
	LLM - RoBERT	0.91	0.96
	LLM - distilBERT	0.91	0.96

Table: **Model Results**

# Results In chart



# Results In chart



## ● Statistical Models

- Random Forest and ensemble models (Voting, Stacking) perform well.
- LightGBM and XGBoost show weaker performance, especially with imbalanced data.

## ● LLM

- Large language models (LLMs) demonstrate outstanding performance in sentiment analysis that effectively capture context and emotional nuances in drug review texts.
- RoBERTa delivers the best results, while DistilBERT is a lightweight and efficient alternative.

## ● Balancing Techniques

- Random Oversampling (ROS) performs more consistently and effectively than SMOTE.
- ROS is better suited for text data, avoiding the creation of semantically unnatural synthetic samples.

**Thank you!**  
Questions?