Project Title	**AI Echo: Your Smartest Conversational Partner**
Skills take away From This Project	• **Data Preprocessing & NLP Techniques** • **Exploratory Data Analysis (EDA)** • **Machine Learning & Deep Learning Models** • **Model Evaluation Metrics** • **Deployment & Visualization**
Domain	**Customer Experience & Business Analytics**

1. Problem Statement

Sentiment analysis is a natural language processing (NLP) technique used to determine the sentiment expressed in a given text. This project aims to analyze user reviews of a ChatGPT application and classify them as positive, neutral, or negative based on the sentiment expressed. The goal is to gain insights into customer satisfaction, identify common concerns, and enhance the application's user experience.

---

2. Business Use Cases

- **Customer Feedback Analysis:** Understand customer opinions to improve product features.
- **Brand Reputation Management:** Monitor user sentiment over time to assess overall perception.

- **Feature Enhancement:** Identify areas for improvement based on negative and neutral reviews.
- **Automated Customer Support:** Prioritize customer complaints based on sentiment classification.
- **Marketing Strategy Optimization:** Develop better engagement strategies based on sentiment insights.

Data Preprocessing

- Removing special characters, stopwords, and punctuation
- Tokenization and lemmatization
- Handling missing values
- Language detection (if applicable)
- Converting text to lowercase

3. Approach.

1. **Data Preprocessing:**
    - Clean and normalize text (removal of stopwords, punctuation, special characters, and stemming/lemmatization).
    - Handle missing values and balance the dataset for unbiased model training.
2. **Exploratory Data Analysis (EDA):**
    - Identify trends in sentiment distribution.
    - Visualize word frequency using word clouds and histograms.
3. **Sentiment Classification Model:**
    - Convert text into numerical features using TF-IDF, Word Embeddings, or Transformer-based embeddings (BERT, GPT, etc.).
    - Train models such as Naïve Bayes, Logistic Regression, Random Forest, LSTMs, or Transformer-based architectures.
4. **Model Evaluation:**
    - Use accuracy, precision, recall, F1-score, and AUC-ROC to assess model performance.
5. **Deployment & Visualization:**
    - Deploy a web-based dashboard using Streamlit or Flask to visualize sentiment trends.

4. Results

- **Sentiment Distribution:** Breakdown of positive, neutral, and negative reviews.
- **Feature Importance:** Key words and phrases influencing sentiment classification.
- **Accuracy Metrics:** Performance comparison of different sentiment classification models.
- **Insights & Recommendations:** Actionable suggestions for application improvements based on analysis.
- **Predict Sentiment** – Classify user reviews into **Positive, Neutral, or Negative** categories.

---

5. Project Evaluation Metrics

- **Accuracy:** Measures the correctness of sentiment classification.
- **Precision & Recall:** Ensures the reliability of positive and negative classifications.
- **F1-Score:** Balances precision and recall to evaluate model performance.
- **Confusion Matrix:** Visual representation of classification errors.
- **AUC-ROC Curve:** Measures the ability of the model to differentiate sentiment categories.

---

6. Technical Tags

- **Technologies:** Python, NLP, Machine Learning, Deep Learning
- **Libraries:** Pandas, NLTK, Scikit-learn.
- **Deployment:** Streamlit, AWS**(optional)**

Data Set:

- **Link : chatgpt_reviews**

Data Set Explanation:

date:
The date when the user submitted the review. It helps analyze trends over time, such as how user sentiment changes with updates.
title:
A short headline summarizing the user's review (e.g., "Great tool!", "Needs improvement"). Useful for quick sentiment cues.
review:
The full written feedback provided by the user. This is the main text body that can be analyzed for sentiment, keyword frequency, or topic modeling.

rating:
A numerical score from 1 to 5 given by the user.

- 1 = very poor
- 5 = excellent
  It reflects the overall satisfaction of the user with ChatGPT.

username:
A randomly generated name representing the reviewer. It gives identity to each review and is useful for identifying repeat users or patterns.

helpful_votes:
The number of other users who found this review helpful. A high number may indicate that the review is well-written or aligns with common opinions.

review_length:
The number of characters in the review text. Longer reviews might be more detailed; shorter ones might be more emotional or blunt.

platform:
Indicates where the user accessed ChatGPT — typically either "Web" or "Mobile". This can help analyze platform-specific feedback or issues.

language:
The language in which the review is written, shown in standard ISO language codes (e.g., "en" for English, "es" for Spanish). Helps with localization analysis.

location:
The country from which the user submitted the review (e.g., USA, India, UK). Useful for regional feedback and market-specific analysis.

version:
The version of ChatGPT the review is about (e.g., "3.5", "4.0"). Helps track user satisfaction across software updates or iterations.

verified_purchase:
Indicates whether the user was a paying or verified subscriber when posting the review. "Yes" means verified, "No" means possibly a free/trial user. It helps validate trust in the review.

Project Deliverables

- Cleaned & Preprocessed Dataset
- EDA Report with Visualizations
- Trained Machine Learning/DL Model for Sentiment Analysis
- Web Dashboard for Sentiment Insights
- Model Performance Report & Insights Document
- Deployment & API Integration (if applicable)

📊 1. What is the distribution of review ratings?

Visualization: Bar chart (1 to 5 stars)
Insight: Understand overall sentiment — are users mostly happy or frustrated?

👍👎 2. How many reviews were marked as helpful (above a certain threshold)?

Visualization: Thumbs up/down count or pie chart
Insight: See how much value users find in reviews, e.g., reviews with more than 10 helpful votes.

🎡 3. What are the most common keywords in positive vs. negative reviews?

Visualization: Two Word Clouds (one for 4–5 stars, one for 1–2 stars)
Insight: Discover what users love or complain about.

📆 4. How has the average rating changed over time?

Visualization: Line chart with `date` on x-axis, average rating on y-axis
Insight: Track user satisfaction over weeks/months.

🌍 5. How do ratings vary by user location?

Visualization: Bar chart or world map
Insight: Identify regional differences in satisfaction or experience.

🧒💻 6. Which platform (Web vs Mobile) gets better reviews?

Visualization: Grouped bar chart comparing average ratings by platform
Insight: Helps product teams focus improvements.

✅❌ 7. Are verified users more satisfied than non-verified ones?

Visualization: Pie chart or side-by-side bar chart comparing rating averages
Insight: Indicates whether loyal/paying users are happier.

🔠 8. What's the average length of reviews per rating category?

Visualization: Box plot or bar chart
Insight: Shows whether people write longer reviews when they're unhappy or very happy.

💬 9. What are the most mentioned words in 1-star reviews?

Visualization: Word cloud or bar chart of top terms
Insight: Spot recurring issues or complaints.

📱🚀 10. What ChatGPT version received the highest average rating?

Visualization: Bar chart (version vs. average rating)
Insight: Evaluate improvement or regression across updates.

Project Guidelines

- Follow best practices for data preprocessing and NLP feature engineering.
- Experiment with different sentiment classification models and compare results.
- Ensure model interpretability and deployability.
- Regularly validate and update the model with new review data.
- Provide actionable recommendations based on sentiment insights.

Key Questions for Sentiment Analysis:(Show in streamlit)

1.What is the overall sentiment of user reviews?

→ Classify each review as **Positive**, **Neutral**, or **Negative**, and compute their proportions.

2.How does sentiment vary by rating?

→ Do 1-star reviews always contain negative sentiment? Is there any mismatch between ratings and actual text?

3.Which keywords or phrases are most associated with each sentiment class?

→ Use word clouds or keyword frequency tables per sentiment type.

4.How has sentiment changed over time?

→ Analyze sentiment trends by month or week to spot peaks in satisfaction or dissatisfaction.

5.Do verified users tend to leave more positive or negative reviews?

→ Compare sentiment distribution between verified_purchase = Yes vs. No.

6.Are longer reviews more likely to be negative or positive?

→ Compare average sentiment scores with review length.

7.Which locations show the most positive or negative sentiment?

→ Helps uncover region-based user experience issues or appreciation.

8.Is there a difference in sentiment across platforms (Web vs Mobile)?

→ Identify where the user experience might need improvement.

9.Which ChatGPT versions are associated with higher/lower sentiment?

→ Determine if a version release impacted user satisfaction.

10.What are the most common negative feedback themes?

→ Use topic modeling or keyword grouping to identify recurring pain points in negative reviews.