# PROJECT IQ ADMIN DASHBOARD

---

**Admin Dashboard for Insights: Top Topics, Concerns, and Recommendations**

---

## Objective

The primary objective of this project is to design and implement a robust system that analyzes user interactions and feedback to generate actionable insights. The dashboard provides real-time information about:

1. The most discussed topics.
2. User concerns derived from negative conversations..
3. Recommendations based on the concerns.

This enables stakeholders to make data-driven decisions, address user pain points, and improve overall service quality.

---

## System Overview

### 1. Data Source

The data is retrieved from a database table containing user interactions (django_message). The relevant fields include:

- prompt: User input or feedback prompt.
- response: System-generated response.
- chat_precursor: Context of the interaction.
- chat_id: Unique identifier for each chat.
- created_at: Timestamp of the interaction.

### 2. Data Preprocessing

- **Libraries Used**:
  - **spaCy**: Tokenization, stopword removal, and lemmatization.
  - **Sentence-BERT**: Embedding generation for semantic understanding.
- **Process**:
  - Text is lowercased and processed to remove stopwords and punctuation.

○ Lemmatization is applied to normalize words to their base form.

### 3. Embedding and Clustering

● **Sentence-BERT**: Converts processed text into dense vector embeddings for semantic analysis.
● **KMeans Clustering**: Groups embeddings into 5 clusters to identify patterns in user feedback.

### 4. Topic Modeling

● **CountVectorizer**: Converts processed text into a term-document matrix.
● **Latent Dirichlet Allocation (LDA)**:
  ○ Generates 5 key topics using probabilistic modeling.
  ○ Identifies top keywords for each topic.
● **Keyword Matching**:
  ○ Matches extracted keywords with predefined topics in an external CSV file (IQ_topics.csv).

### 5. Sentiment Analysis

● **TextBlob**: Analyzes sentiment polarity:
  ○ Positive: User satisfaction.
  ○ Negative: User concerns.
  ○ Neutral: Neutral feedback or context.

### 6. Insights Generation

● Identifies the most frequent topics.
● Highlights concerns derived from negative sentiment.
● Displays top keywords for each topic and matches them with predefined categories.

### 7. Sample Concerns

For each topic, negative feedback is summarized with up to 3 sample concerns to provide qualitative insights.

# Technical Implementation

**Tools and Technologies**

- **Programming Language**: Python
- **Libraries**:
    - Data Handling: Pandas, NumPy
    - NLP: spaCy, Sentence-BERT, TextBlob
    - Machine Learning: scikit-learn (KMeans, LDA)
    - Visualization: Integrated into the admin dashboard UI
- **Database**: SQL-based backend for data retrieval
- **External Resources**:
    - IQ_topics.csv: Predefined topics for keyword matching.

---

# Pipeline Workflow

**Step 1: Data Retrieval**

- SQL query fetches data from the database.

**Step 2: Text Preprocessing**

- Text is processed to remove noise, normalize language, and prepare for further analysis.

**Step 3: Clustering**

- Semantic embeddings are generated for feedback, and KMeans groups these into 5 clusters.

**Step 4: Topic Modeling**

- LDA identifies dominant themes across feedback data.
- Keywords are extracted for each topic and matched with predefined categories.

**Step 5: Sentiment Analysis**

- Each feedback entry is scored for sentiment polarity.
- Entries with negative sentiment are flagged as concerns.

**Step 6: Insights Aggregation**

- Topics are ranked by frequency.
- Negative feedback is grouped under matched topics for concern analysis.
- Example concerns are displayed for each high-frequency topic.

---

# Outputs

**Dashboard Insights**

1. **Top Topics**:
   - Frequency of each topic based on LDA analysis and clustering.
   - Keywords representing the themes of each topic.
2. **Concerns**:
   - Count of negative feedback grouped by topic.
   - Sample concerns provided for context.
3. **Recommendations**:
   - Suggestions for addressing user concerns based on extracted topics.

**Data Summary:**

- **Columns**:
  - combined_text: Original text.
  - processed_text: Preprocessed version of text.
  - cluster: Cluster assignment by KMeans.
  - lda_topic: Topic index by LDA.
  - topic_keywords: Keywords for each topic.
  - matched_topic: Predefined topic match.
  - sentiment: Sentiment polarity.
- **Final DataFrame**: Aggregated data for visualization, including topic, concern_text, and concern_count.

---

# Impact

This system provides valuable insights into user behavior and feedback, empowering decision-makers to:

1. Prioritize resources based on user concerns.
2. Identify and address pain points to enhance user experience.
3. Leverage data to improve service quality and satisfaction.