Real-Time Customer Sentiment Analysis

Group Members: Jeremy Granizo, Mahmoud Saad
Course: CS482 (Data mining)
Instructor: Khalid Bakhshaliyev

Project Overview: We propose to develop a real-time entity-specific sentiment analysis system using data mining and natural language processing techniques. The goal is to accurately identify entities mentioned in social media posts, particularly tweets, and determine the sentiment expressed towards each entity. By fine-tuning small-scale foundation language models (LLMs) like DistilBERT or TinyBERT on a cleaned and preprocessed Twitter dataset, we aim to achieve high accuracy in aspect-based sentiment classification. This system will enable businesses to gain granular insights into public perception of specific products, services, or brands.

Potential Datasets:

Starting Dataset:

- **Sentiment140 Dataset**
    - Description: This dataset contains 1.6 million tweets labeled as positive, negative, or neutral. It's easy to work with, and since it's relatively clean, it will allow us to focus on building and evaluating our sentiment analysis model quickly.
    - Features:
        - Sentiment: Sentiment of the tweet (0 = Negative, 4 = Positive).
            - This will be our Y or **target** variable.
        - Id: Unique ID of the tweet.
        - Date: Date and time of the tweet.
            - The date is stored in this format "*Sat May 16 23:58:44 UTC 2009*".
        - Query: Query used to obtain the tweet.
            - This column might be deleted since the only unique value in this column is "NO_QUERY"
        - User: Username of the person who tweeted.
        - Text: The actual tweet text.
    - Link: Sentiment140 dataset with 1.6 million tweets
    - File Format: CSV
    - Size: ~1.6 million tweets
    - Tutorial: Tutorial on how to get Data from twitter and train a model using the Sentiment140 dataset

- Preliminary work:
  - We need to remove the query column since it won't have an impact on the model.
  - Setting the positive tweet value to 1 will be also helpful, so we can use binary models having the negative tweet value = 0.
  - We need to filter the text to make it easier for our model to detect words that relate to a sentiment.
  - We will be removing the links from the text.
  - Lower Casing: Each text is converted to lowercase.
  - Replacing URLs: Links starting with "http" or "https" or "www" are replaced by "URL".
  - Replacing Emojis: Replace emojis by using a pre-defined dictionary containing emojis along with their meaning. (eg: ":)" to "EMOJIsmile")
  - Replacing Usernames: Replace @Usernames with the word "USER". (eg: "@Kaggle" to "USER")
  - Removing Non-Alphabets: Replacing characters except Digits and Alphabets with space.
  - Removing Consecutive letters: 3 or more consecutive letters are replaced by 2 letters. (eg: "Heyyyy" to "Heyy")
  - Removing Short Words: Words with a length of less than 2 are removed.
  - Lemmatizing: Lemmatization is the process of converting a word to its base form. (e.g: "Great" to "Good")
- Citation: *Go, A., Bhayani, R. and Huang, L., 2009. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, 1(2009), p.12*

- **Amazon Product Sales Dataset**
  - Description: Amazon Product Data separated by 142 categories scraped from the Amazon website.
  - Features:
    - Name
      - The name of the product
    - Main_category
      - Initial main category of the product
        - This will be our Target variable
    - Sub_category
      - The main category of the product belonging to under the initial main category
    - Image
      - The image of the product look like
    - Link
      - The amazon website reference link of the product
    - Ratings
      - The ratings given by amazon customers of the product
    - Number of Ratings

            - The number of ratings given to this product in amazon shopping
        - Discount price
            - The discount prices of the product
        - Actual Price
            - Actual maximum retail price
    - Link: https://www.kaggle.com/datasets/lokeshparab/amazon-products-dataset
    - File format: Csv file
    - Size~ 300,000 products
    - Preliminary work:
        - Replace null values in certain columns
        - Lower casing all columns with text values

Fine-Tuning for Entity-Specific Sentiment Analysis:

- **Entity Twitter Sentiment Analysis Dataset**
    - Entity-level sentiment analysis on multi-lingual tweets.
    - Description: This is an entity-level sentiment analysis dataset of twitter. Given a message and an entity, the task is to judge the sentiment of the message about the entity. There are three classes in this dataset: Positive, Negative and Neutral. We regard messages that are not relevant to the entity (i.e. Irrelevant) as Neutral.
    - Features:
        - Sentiment: Sentiment of the tweet (Negative, Positive, Irrelevant, Neutral).
            - This will be our Y or **target** variable.
        - Tweet Id: Unique ID of the tweet.
        - Entity: Date and time of the tweet.
        - Tweet content: The actual tweet text.
    - Dataset Link:Twitter Sentiment Analysis | Kaggle

Project Enhancement:

Simulating Real-Time Analysis (Optional)

- **Data Streaming Simulation:**
    - Use time-stamped tweets to simulate real-time data flow.

    - Analyze how the model performs over time and how sentiment trends evolve.
- **Visualization:**
    - Create dashboards displaying entity-specific sentiments in real-time.
    - Visualize sentiment trends for specific entities over the simulated time period.

System Architecture:

1. **Data Preprocessing**: Clean and preprocess tweets by removing noise (e.g., links, special characters).
2. **Model Training**:
    - Train a baseline sentiment analysis model using Sentiment140.
    - Fine-tune on the entity-specific dataset to enhance contextual sentiment understanding.
3. **Real-Time Data Pipeline** (Optional):
    - Use the **Twitter API** and **Apache Kafka** to ingest live tweets.
    - Apply trained models to classify sentiments in real-time.
4. **Recommendation System** (Future Enhancement):
    - Use Amazon Product Reviews for building a recommendation engine.
    - Link live sentiments to the recommendation engine to prioritize relevant products.

Resources:

https://medium.com/@manjindersingh_10145/sentiment-analysis-with-bert-using-huggingface-88e99deeec9a(using Bert for sentiment analysis)