Bachelor of Computing Honours in Computer Science
Bachelor of Computing Honours in Information Systems
CCS2062 | CSE2062 - Artificial Intelligence

The project report on

Sentiment Analysis

GitHub link: https://github.com/RudraharanNivaethan/sentiment-analyzer-webapp.git

Submitted By

Rudraharan Nivaethan – FC222008
Nanthakumar Vepusanan – FC221023
Baskaran Natheesan – FC221014

Under the guidance of:

Dr. Buddhitha Hettige

UNIVERSITY OF SRI JAYEWARDENEPURA

Faculty of Computing

Authors:

Rudraharan Nivaethan
Index Number: FC222008
Registration Number: FC115565
Department of Software Engineering
Faculty of Computing
University of Sri Jayewardenepura
Nugegoda, Sri Lanka
nivaethanrudra6@gmail.com

Nanthakumar Vepusanan
Index Number: FC221023
Registration Number: FC115287
Department of Computer Science
Faculty of Computing
University of Sri Jayewardenepura
Nugegoda, Sri Lanka
vepusanan@gmail.com

Baskaran Natheesan
Index Number: FC221014
Registration Number: FC115266
Department of Computer Science
Faculty of Computing
University of Sri Jayewardenepura
Nugegoda, Sri Lanka
natheesanbaskaran@gmail.com

1. Introduction

In a world fueled by data, sentiment analysis of textual data is an ever-important task for businesses, researchers, and content creators. Given the immense volume of text data generated daily from social media, customer reviews, survey responses, and online discussion forums, completing a manual sentiment analysis is neither time efficient nor practical. In essence, the Sentiment Analysis web application is an artificial intelligence (AI) driven application that can automatically analyze and assess the sentiment and subjectivity of text input. The application employs Natural Language Processing (NLP) methods to analyze text and generate numerous outputs that include compound sentiment scores, overall sentiment classification, consideration of subjectivity, and a breakdown of sentiment by section or text component. This project illustrates a highly relevant and meaningful use of Artificial Intelligence and Natural Language Processing technologies to address real-world challenges. By utilizing two widely used NLP tools VADER (Valence Aware Dictionary and sEntiment Reasoner) and TextBlob the application provides a multi-faceted sentiment analysis that considers emotional polarity (positive versus negative) as well as level of subjectivity. The web interface provides ease of use for various audience segments that may include social media analysts, customer feedback analysts, and educators seeking to provide demonstrations.

Built with Flask, NLTK, TextBlob, Matplotlib, and WeasyPrint, the application offers real-time analysis, interactive visualizations, and exportable PDF reports, making sentiment insights both interpretable and portable for various use cases.

2. Problem Statement

Organizations and individuals encounter enormous challenges when attempting to assess the large amounts of textual data produced each day. Manually analyzing customer reviews, social media posts, survey responses, and feedback is tedious, time-consuming, and biased. Accordingly, there are three primary issues that led to this project:

2.1 Volume and Scale

The volume of user-generated content makes manual sentiment analysis impractical. Organizations receive thousands of customer reviews, social media mentions, and other feedback messages that require speedy processing to maintain decision-making.

2.2 Objective and Consistent

Human interpretation of sentiment will always be subjective and not achieve consistent outcomes from analyst to analyst. Individuals may interpret the same body of text in different ways due to personal biases, cultural background, or level of contextual awareness.

2.3 Dimensionality Limitation
Traditional sentiment analysis tends to only look at positive or negative classification and misses the critical importance of the body of text being factual (objective) or opinioned (subjective). Limiting insight and assessment available for decision-making.

2.4 Accessibility of NLP tools
There are great NLP libraries available, however, they typically require some level of technical expertise to work with partially or effectively. Non-technical users in marketing, customer service, research fields, etc., have difficulty finding tools that manage sentiment analysis.

2.5 Real-Time Analysis Requirements
Numerous applications, such as measuring brand reputation during a product launch or gauging sentiment during crisis management, benefit from immediate insight rather than delayed batch processing.

Project Goals
The goal of thins project is to build a user-friendly, AI-based web application that provides:
- Automated real-time sentiment analysis that is available for non-technical users
- Multi-dimensional insight with respect to emotional polarity (positive/negative/neutral) and subjectivity (objective/opinionated)
- Visualizations of sentiment distributions for ease of use
- Reports that can be exported and documented
- A consistent and unbiased analysis based on pre-established NLP models

3. Development

3.1 System Architecture
The application design is based on a modular three tier architecture:
Frontend: An intuitive interface for the application using HTML, CSS, and JavaScript by creating 4 pages (Home, App, About, Contact). JavaScript allows the user to dynamically render content to the page without requiring the page to be refreshed.
Backend: Flask works to manage the routing and requests coming from the user to the corresponding modular service components.
AI/NLP Layer: Three separate services for each analysis, which takes care of sentiment analysis (VADER + TextBlob), chart generation (Matplotlib), and then generating a PDF report (WeasyPrint).

3.2 AI Models
VADER: Lexicon-based approach optimized for use on social media text, emojis, and slang. It provides output in the form of percentages of positive, neutral, and negative and a compound score (between -1 and +1).
TextBlob: Like VADER, but it observes degrees of subjectivity (0 = objective, 1 = subjective). TextBlob distinguishes between factual based statements and opinion-based statements.
Integration: VADER is used to provide emotional tone of the statement and then TextBlob is used to provide degrees of objectivity. Based on the percentages from TextBlob we classify text as Objective (≤0.4), Moderately Subjective (0.4-0.7), and Highly Subjective (>0.7).

3.3 Technical Implementation

Data Flow: A publication takes the user's text and processes it using a sentiment_service (VADER + TextBlob) and then the result is visualized using Matplotlib charts, which are converted to images using Base64 encoding, thus not creating temporary files. In the document generation step, WeasyPrint creates complete PDF reports with embedded charts, instead of temporarily saving images and including the file path in the document.

Best Practices: Modular code architecture, separation of concerns, isolated variables, complete documentation, good attention to version control hygiene, modular naming conventions, and dependency isolation via virtual environments are best practices implemented in the project.

4. Evaluation

4.1 Functional Testing

Testing variations across samples confirmed operational capacity on a number of fronts:

- Social media: Accurately classified sentiment in informal posts with emoji and slang
- Customer Reviews: Accurately identified classification of positive, negative, or mixed
- Objective Content: Accurately categorized factual statements with low subjectivity scores

4.2 Performance Review

Strengths: Using dual models provides results for emotional tone as well as objectivity levels. VADER also accounts for intensity modifiers, negation, and emphasis. Fast enough for real time usage (ms).

Limitations: Lexicon based method may inaccurately classify sarcasm or irony. Classifying "mixed sentiment" sentences displayed averaged capabilities based on the aggregated working model. English-only coding constrains applications for multilingual market opportunities.

4.3 User Experience

The interface, through careful labeling (e.g. emoji indicators), easy-to-use input form, immediate visual response, and ease of use via a single PAGE flow, mostly achieves a level of accessibility for inputting users with minimal to no tech ability. The option to export identification via PDF also allows for efficient documentation and sharing.

4.4 Achievements of Objective

Achievement of objective provided: automated identification; a multi-dimensional approach to understanding polarity or subjectivity; visual representation through charts; PDF report option; reliability in results; an interface usable for non-tech processes. In effect, the usability of the interface democratized NLP potential.

5. Conclusion

The Sentiment Analysis Web App effectively showcases an applied approach to AI/NLP applicable to text analysis. The combination of VADER and TextBlob allows for other insights beyond categorical positive and negative classifications.

Key Outcomes: The web application has demonstrated functional value for non-technical users, a dual model approach to NLP, real-time speed with interactive visualizations, a modular codebase following best practices, and a report generator capable of including visualizations in a sharable form.

Practical Considerations: The application is applicable to businesses (customer sentiment), researchers (surveys), educators (showcasing NLP), and content creators (audience reaction). Utilization of easy-to-use tools is attractive because it still yields fast results without expensive computational resources.

Limitations: The application has difficulty adequately comprehending sarcasm and irony, it only works in English, and subcategories of subjectivity are too limited. All three limitations are acceptable, when the application is used for education or demonstration.

Future Improvements: Possible future improvements may include deep learning models (BERT, RoBERTa) for an improved understanding of context, multilingual capabilities, domain-specific dictionaries, integration with a database for batch computing, and an interactive dashboard for longer-term visualizations.

Our project demonstrated the accessibility of modern NLP tools, when used thoughtfully. This application demonstrates usability, clear visualizations, and full report generation that converts complex capabilities of AI to an easy-to-use tool that provides immediate value in educational, business, or research contexts.