Lexicon-Based Sentiment Analysis of Movie Reviews

Jude Eschete
Dept. of Computer Engineering
Stevens Institute of Technology
Hoboken, NJ, USA
jeschete@stevens.edu

Ella Disanti
Dept. of Computer Engineering
Stevens Institute of Technology
Hoboken, NJ, USA
edisanti@stevens.edu

Raymond Donkemezuo
Dept. of Computer Engineering
Stevens Institute of Technology
Hoboken, NJ, USA
rdonkemez@stevens.edu

***Abstract*—This paper presents a lexicon-based sentiment analysis approach to classifying movie reviews using the Large Movie Review Dataset (IMDb). We investigate several lexicons, design a rule-based algorithm to account for negation, and implement key functions in Python. Our system preprocesses text, handles contextual polarity inversion, and classifies reviews as positive, negative, or neutral based on a custom scoring aggregation mechanism.**

I. Background

Sentiment analysis is a subfield of Natural Language Processing (NLP) aimed at extracting subjective information from textual data. Lexicon-based approaches rely on predefined dictionaries of words associated with sentiment scores. These methods are interpretable and lightweight compared to machine learning models but can be limited by context and domain-specific language. Our project builds a robust lexicon-based sentiment classifier, tailored for movie review data.

II. System Architecture

Our system is structured around a custom Python class, `MovieSentimentAnalyzer`. It consists of several modular components:

- Data loading from a CSV file
- Text preprocessing (cleaning, tokenization, stop word removal, stemming)
- Negation handling with scope-based polarity inversion
- Lexicon-based score aggregation
- Sentiment classification logic

This modular design enables clean integration with datasets and easy extensibility for future improvements.

III. Design and Algorithms

We implemented the following methods:

- **`load_kaggle_data(path)`**: Loads the IMDb dataset.
- **`preprocess_text(text)`**: Cleans and tokenizes input, removes stop words, applies stemming.
- **`apply_negation_handling(tokens)`**: Detects negation terms and marks affected words.
- **`compute_sentiment_score(tokens)`**: Aggregates word-level scores with negation adjustments.
- **`classify_sentiment(score)`**: Maps numerical score to a polarity label.
- **`analyze_review(text)`**: Executes the full analysis pipeline.

The negation handling method tags tokens within a 3-word window after negators such as `''not''` or `''never''`, inverting their sentiment scores if present in the lexicon.

IV. Testing and Simulation Results (In Progress)

Initial tests on a small subset of the IMDb dataset indicate that the system is functioning correctly in terms of classification logic. Reviews with clear positive or negative language are accurately scored. Ongoing work includes evaluating overall accuracy using labeled data and refining the lexicon for better domain coverage.

V. Conclusions (In Progress)

Thus far, our lexicon-based sentiment analysis system shows promise for classifying movie reviews with minimal computational overhead. The modular design facilitates experimentation with various lexicons and preprocessing techniques. Final evaluation metrics will be provided upon full dataset integration and testing.

References

[1] A. L. Maas et al., ”Learning Word Vectors for Sentiment Analysis,” in *Proc. of the 49th Annual Meeting of the Association for Computational Linguistics*, 2011.

[2] O. Bulut, ”Lexicon-Based Sentiment Analysis Using R,” Medium, 2024.

[3] S. Narayanan, ”Sentiment Analysis in NLP: Key Techniques and Insights,” Sapien.io, 2025.

[4] P. Turney, ”Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews,” *ACL*, 2002.

[5] Y. Liu et al., ”Sentiment Analysis and Opinion Mining,” *Synthesis Lectures on Human Language Technologies*, 2012.