Natural Language Opinion Search Engine Report

Joshua Gulizia
2269002
University of Houston
`Natural Language Processing`

1 Overview

The proliferation of online product reviews has created both opportunities and challenges for consumers seeking specific information about product features. While the abundance of user-generated content provides valuable insights, finding reviews that discuss particular aspects with specific sentiments remains difficult. This project addresses this challenge by developing a Natural Language Opinion Search Engine that retrieves reviews based on aspect-opinion pairs.

The system processes queries in the format "aspect:opinion" (e.g., "audio quality:poor") and returns reviews that not only mention the aspect but express the specified sentiment. This approach goes beyond traditional keyword matching by understanding the relationship between product features and their evaluative expressions. The engine implements three distinct retrieval methods, each building upon natural language processing techniques learned throughout the course.

The primary contributions of this project include:

- Implementation of a Boolean baseline system for comparative evaluation
- Development of an enhanced retrieval method combining star ratings with opinion lexicons
- Creation of a semantic search approach using modern sentence embeddings
- Comprehensive evaluation demonstrating significant improvements over baseline approaches

2 Background

2.1 Information Retrieval in E-commerce

The digital marketplace has fundamentally transformed how consumers make purchasing decisions. Product reviews have become crucial sources of information, with studies showing that 93% of consumers read online reviews before making purchases [1]. However, the sheer volume of reviews creates an information overload problem. A popular product on Amazon may have thousands of reviews, making it impractical for consumers to read them all.

Traditional search engines use Boolean retrieval models that match exact keywords but fail to understand context or sentiment. For instance, searching for "battery life" might return reviews mentioning these words without distinguishing between positive ("excellent battery life") and negative ("poor battery life") contexts. This limitation motivates the need for opinion-aware search systems.

2.2 Opinion Mining and Sentiment Analysis

Opinion mining, also known as sentiment analysis, involves automatically identifying and extracting subjective information from text. In the context of product reviews, this includes determining:

- **Aspects**: The specific features or components being discussed (e.g., "screen quality," "customer service")
- **Opinions**: The sentiment expressed toward these aspects (e.g., "excellent," "terrible")
- **Polarity**: Whether the overall sentiment is positive, negative, or neutral

Previous work in this area includes Liu's aspect-based sentiment analysis [2] and Hu and Liu's opinion lexicon [1], which provides comprehensive lists of positive and negative opinion words frequently used in reviews.

2.3 Course Foundations

This project builds upon several key concepts from the Natural Language Processing course:

- **Text Preprocessing**: Tokenization, stop word removal, and lemmatization
- **Information Retrieval**: Boolean search and inverted index construction
- **Language Models**: N-gram analysis and semantic representations
- **Text Classification**: Using supervised learning for sentiment prediction
- **Parsing**: Understanding grammatical relationships between words

3 Method

3.1 Problem Formulation

I formalize the opinion search problem as follows: Given a query $q = \{aspect : opinion\}$ where the aspect contains at most two words and the opinion contains one or two words, retrieve all reviews R that discuss the specified aspect with the indicated sentiment orientation.

The retrieval task must address several challenges:

1. **Aspect Identification**: Correctly identifying when a review discusses the queried aspect
2. **Opinion Association**: Determining if expressed opinions relate to the specific aspect
3. **Polarity Matching**: Ensuring retrieved reviews express the correct sentiment polarity

3.2 Dataset

I utilized a real-world corpus of 210,761 Amazon product reviews covering electronic and software products. The dataset includes:

- Review text content
- Star ratings (1-5 scale): 1 Star (28,611), 2 Star (15,561), 3 Star (19,224), 4 Star (38,709), 5 Star (108,656)
- Review metadata (title, helpfulness votes, etc.)
- 50 major topic clusters representing common discussion themes

3.3 Preprocessing Pipeline

My preprocessing pipeline transforms raw review text into a searchable format:

Algorithm 1 Text Preprocessing Pipeline

```
1: Input: Raw review text
2: Convert text to lowercase
3: Handle emoticons: ":)" → "positive_emoji", ":(" → "negative_emoji"
4: Remove special characters (keep alphanumeric and spaces)
5: Tokenize into words
6: Remove stop words from curated list
7: Filter words appearing less than 5 times
8: Output: Cleaned token list
```

4 Design

4.1 System Architecture

The search engine consists of three main components working in sequence:

1. **Query Parser**: Processes user queries to extract aspect and opinion terms
2. **Retrieval Engine**: Implements three different search strategies
3. **Result Ranker**: Orders results based on relevance scores

4.2 Baseline: Boolean Search

The baseline system implements traditional Boolean retrieval using an inverted index. For each query, I evaluate three search strategies:

- **Test 1**: Retrieve reviews containing at least one aspect word
- **Test 2**: Retrieve reviews containing both aspect AND opinion terms
- **Test 3**: Retrieve reviews containing aspect OR opinion terms

4.3 Method 1: Rating and Lexicon Enhanced Search

This approach enhances Boolean search by incorporating:

1. **Opinion Lexicon Integration**: I utilize Hu and Liu's opinion lexicon containing 2,006 positive and 4,783 negative words to determine opinion polarity.
2. **Rating-Based Filtering**: Reviews are filtered based on star ratings:
    - Positive opinions: Include only reviews with rating > 3
    - Negative opinions: Include only reviews with rating ≤ 3
3. **Lexicon Expansion**: For queries with limited results, I expand opinion terms with synonyms from the lexicon.

4.4 Method 2: Semantic Search with Sentence Embeddings

My most advanced method employs sentence transformers to capture semantic similarity:

Algorithm 2 Semantic Search Algorithm

```
Extract candidate reviews containing aspect terms
For each candidate review:
    Extract sentences mentioning aspects
    Compute sentence embeddings using SBERT
    Calculate semantic similarity with query
    Apply polarity consistency check
Return top-k reviews by combined score
```

4.5 Implementation Challenges

Due to time constraints from managing a full-time internship alongside four courses, several implementation decisions were made to balance functionality with feasibility:

- Limited semantic search to 1,000 candidates to manage computational resources
- Used pre-trained models rather than fine-tuning on domain-specific data
- Implemented approximate rather than exhaustive grammatical parsing

5 Results

5.1 Evaluation Methodology

I evaluated the system using five test queries representing common product review scenarios:

1. "audio quality:poor" - Negative audio-related opinions
2. "wifi signal:strong" - Positive connectivity opinions
3. "mouse button:click problem" - Negative usability opinions
4. "gps map:useful" - Positive functionality opinions
5. "image quality:sharp" - Positive visual quality opinions

Precision was calculated as: $Precision = \frac{\text{\# Relevant Retrieved}}{\text{\# Total Retrieved}}$

5.2 Performance Results

Table 1 presents the evaluation results across all methods:

Table 1: Retrieval Performance Comparison

Query	Baseline (Boolean)			Method 1 (M1)			Method 2 (M2)		
	#Ret	#Rel	Prec	#Ret	#Rel	Prec	#Ret	#Rel	Prec
audio quality:poor	714	321	0.450	468	312	**0.667**	100	65	0.650
wifi signal:strong	892	401	0.449	145	89	0.614	100	63	0.630
mouse button:click problem	556	234	0.421	312	198	0.635	100	62	0.620
gps map:useful	423	198	0.468	147	96	0.653	100	64	0.640
image quality:sharp	687	316	0.460	234	156	**0.667**	100	66	0.660
Average	654.4	294.0	0.450	261.2	170.2	**0.647**	100	64.0	0.640

5.3 Key Findings

1. **Precision Improvement**: Method 1 achieved the highest average precision (64.7%), representing a 44% improvement over the baseline.
2. **Retrieval Reduction**: Advanced methods retrieved fewer but more relevant documents, with Method 2 limiting results to 100 high-confidence matches.
3. **Query Sensitivity**: Performance varied by query type, with "audio quality:poor" achieving the highest precision (66.7%) in Method 1.

6 Discussion

6.1 Method Effectiveness

The significant improvement in precision from the baseline to advanced methods validates my hypothesis that understanding opinion semantics enhances retrieval quality. Method 1's success can be attributed to:

- **Rating Alignment**: Star ratings provide strong signals for overall sentiment
- **Lexicon Coverage**: The comprehensive opinion lexicon captures diverse expression patterns
- **Balanced Approach**: Combining multiple signals reduces false positives

Method 2's semantic approach, while achieving slightly lower precision than Method 1, offers advantages in:

- Capturing implicit opinions not in the lexicon
- Understanding contextual sentiment expressions
- Providing consistent result sizes for user experience

6.2 Error Analysis

Common sources of retrieval errors I identified included:

1. **Aspect Ambiguity**: Terms like "quality" appear in multiple contexts
2. **Mixed Sentiments**: Reviews discussing multiple aspects with different opinions
3. **Sarcasm and Irony**: "Great quality - if you enjoy static!"
4. **Comparative Statements**: "Better than X but still poor"

6.3 Limitations and Future Work

Time constraints from balancing academic and professional commitments led to several limitations:

- **Limited Evaluation**: Manual evaluation was performed on samples rather than complete result sets
- **Simple Query Format**: Restriction to two-word aspects may miss complex features
- **English-Only**: System currently supports only English reviews
- **Domain Specificity**: Trained on electronics reviews, may not generalize

Future improvements could include:

- Fine-tuning language models on review-specific data
- Implementing aspect-opinion dependency parsing
- Supporting multi-aspect queries
- Creating domain-specific opinion lexicons

7 Conclusion

This project successfully demonstrated that incorporating natural language understanding significantly improves opinion-based review retrieval. My implementation achieved a 44% improvement in precision over traditional Boolean search, validating the importance of semantic understanding in information retrieval tasks.

The development process reinforced key NLP concepts including text preprocessing, sentiment analysis, and semantic similarity. Despite time constraints from concurrent commitments, I delivered a functional system that addresses real-world e-commerce challenges.

This work contributes to the growing field of opinion-aware search systems and provides a foundation for future enhancements. As online reviews continue to proliferate, systems that can intelligently filter and retrieve relevant opinions will become increasingly valuable for both consumers and businesses.

The source code and evaluation data are available at: `https://github.com/jgulizia/nlp-opinion-search`

References

[1] Minqing Hu and Bing Liu. *Mining and Summarizing Customer Reviews*. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2004), Aug 22-25, 2004, Seattle, Washington, USA.

[2] Bing Liu. *Sentiment Analysis and Subjectivity*. Handbook of Natural Language Processing, Second Edition, (editors: N. Indurkhya and F. J. Damerau), 2010.

[3] Hinrich Schütze, Christopher D. Manning, and Prabhakar Raghavan. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[4] Nils Reimers and Iryna Gurevych. *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2019.