SENTIMENT ANALYSIS PROJECT REPORT: CUSTOMER REVIEW CLASSIFICATION USING J48 DECISION TREE BY OREOLUWA ANJORIN

EXECUTIVE SUMMARY

This project successfully implemented a sentiment analysis system for customer reviews using machine learning classification techniques. The primary objective was to develop an automated system capable of accurately classifying product reviews as positive or negative sentiment using the J48 decision tree algorithm in WEKA. The project involved web scraping over 1,000 customer reviews from e-commerce platforms, data preprocessing, and comprehensive model evaluation.

1. Project Objectives

The main objectives of this sentiment analysis project were:

- **Data Collection:** Scrape 1,000+ customer reviews from any platform.

- **Sentiment Classification:** Develop a robust classification model using J48 decision tree algorithm to automatically categorize reviews as positive or negative

- **Model Evaluation:** Assess model performance using both cross-validation and independent test set validation

- **Practical Implementation:** Create a deployable solution for automated sentiment analysis of product reviews

2. Methodology and Data Collection

2.1 Web Scraping Implementation

The data collection phase utilized a custom Python scraping solution targeting multiple product pages on Jumia Nigeria. The scraping strategy employed:

- **Multi-product Approach:** Targeted 5 different product URLs to ensure diverse review content

- **Respectful Scraping:** Implemented 2-second delays between requests to avoid server overload

- **Quality Filtering:** Applied text length filters (20-1000 characters) and keyword-based relevance checks

- **Duplicate Handling:** Used set-based storage to automatically eliminate duplicate reviews

The scraping process successfully collected over 1,100 unique reviews, which were then split into:

- **Training Set:** 1,000 reviews (Nivea_men_reviews.txt)

- **Test Set:** 100+ reviews (nivea_test.txt)

2.2 Data Preprocessing Challenges And Solutions

Critical Data Format Issues

During the WEKA implementation phase, several significant challenges were encountered:

Challenge 1: ARFF File Format Recognition

- Issue: WEKA failed to recognize the Nivea_men_reviews.arff file as a valid ARFF format

- Root Cause: Missing nominal value declarations in the file header and presence of problematic punctuation marks in review text

- Solution: Extensive data cleaning to remove duplicated punctuation marks and special characters that interfered with ARFF parsing

Challenge 2: Text Data Compatibility

- Issue: Raw text reviews required conversion to numerical features for J48 processing

- Solution: Applied StringToWordVector filter in WEKA to transform text data into bag-of-words representation

2.3 Sentiment Labelling Strategy

An automated sentiment labelling approach was implemented using lexicon-based classification:

- Positive Indicators: 'good', 'great', 'excellent', 'amazing', 'perfect', 'love', 'recommend', 'quality'

- Negative Indicators: 'bad', 'terrible', 'poor', 'disappointing', 'problem', 'issue'

- Scoring System: Sentiment score = (positive_word_count - negative_word_count)

- Classification Rules:

  - Score > 0: Positive sentiment

  - Score < 0: Negative sentiment

  - Score = 0: Contextual analysis using text length and word patterns

3. WEKA Implementation Process

3.1 Detailed Implementation Steps

Phase 1: Environment Setup

1. Launched WEKA GUI Chooser and selected Explorer interface

2. Loaded training dataset (Nivea_men_reviews.arff) via Preprocess tab

3. Verified data integrity: 2 attributes (text: String, class: Nominal)

Phase 2: Text Preprocessing

1. Applied StringToWordVector filter under filters → unsupervised → attribute

2. Transformed textual reviews into numerical feature vectors

3. Generated bag-of-words representation suitable for J48 algorithm

Phase 3: Classifier Configuration

1. Selected J48 decision tree classifier from trees → J48

2. Configured parameters:

   - Confidence Factor: 0.25 (standard pruning level)

   - Minimum Objects per Leaf: 2

   - Applied pruning for generalization

Phase 4: Model Training and Testing

1. Cross-Validation Setup: 10-fold cross-validation on training data

2. Independent Test Set: Configured supplied test set using nivea_test.arff

3. Ensured consistent preprocessing by applying same StringToWordVector filter to test data

4. Results and Performance Analysis

4.1 Cross-Validation Results (Training Data)

The J48 classifier demonstrated excellent performance on the training dataset:

- Overall Accuracy: 95.9% (959 out of 1000 instances correctly classified)

- Incorrectly Classified 4.1% (41 instances)

- Kappa Statistic: 0.8423 (substantial agreement)

- Mean Absolute Error: 0.0776

- Root Mean Squared Error: 0.1503

Detailed Performance Metrics:

Class	Precision	Recall	F-Measure	Support
Positive	0.966	0.986	0.976	846 instances
Negative	0.917	0.821	0.866	154 instances
Weighted Average	**0.959**	**0.959**	**0.958**	**1000 instances**

Confusion Matrix Analysis:

- True Positive (Positive correctly classified): 834

- False Negative (Positive misclassified as Negative): 12

- True Negative (Negative correctly classified): 126

- False Positive (Negative misclassified as Positive): 28

4.2 Independent Test Set Results

The model's generalization capability was evaluated using the independent test set:

- Test Set Accuracy: 87.0% (87 out of 100 instances correctly classified)

- Incorrectly Classified: 13.0% (13 instances)

- Model Build Time: 0.51 seconds

- Test Evaluation Time: 0.02 seconds

Test Set Performance by Class:

Class	Precision	Recall	F-Measure
Positive	0.870	1.000	0.930
Negative	N/A*	0.000	N/A*
Weighted Average	**0.870**	**0.870**	**N/A***

Note: The test set showed perfect recall for positive class but zero recall for negative class, indicating class imbalance in the test data.

5. Decision Tree Analysis

5.1 Tree Structure Insights

The J48 algorithm generated a decision tree with the following characteristics:

- Number of Leaves: 55

- Tree Size: 109 nodes

- Build Time: 0.67 seconds

5.2 Key Decision Rules

The decision tree revealed important patterns in review classification:

- Primary Split Features: Text features related to product quality assessments

- Positive Indicators: Presence of words like "better", "used", "just", "Im" showed strong positive sentiment correlation

- Negative Indicators: Specific linguistic patterns associated with dissatisfaction

6. Model Strengths and Limitations

6.1 Strengths

1. **High Training Accuracy:** 95.9% accuracy demonstrates strong pattern recognition

2. **Efficient Processing:** Sub-second model building and prediction times

3. **Interpretable Results:** Decision tree structure provides transparent decision logic

4. **Robust Cross-Validation:** Consistent performance across 10-fold validation

6.2 Limitations and Areas for Improvement

1. **Class Imbalance:** Test set showed significant imbalance between positive and negative reviews

2. **Generalization Gap:** 8.9% accuracy drop from training to test set suggests potential overfitting

3. **Limited Negative Detection:** Poor recall for negative sentiment in test data (0.000)

4. **Text Preprocessing Sensitivity:** ARFF format issues highlighted data quality dependencies

7. Technical Challenges Overcome

7.1 Data Quality Issues

- Successfully resolved ARFF file recognition problems through systematic punctuation cleaning

- Implemented robust duplicate detection and removal processes

- Developed effective text normalization procedures

7.2 Feature Engineering

- Applied appropriate text-to-feature conversion using StringToWordVector

- Maintained consistency between training and test data preprocessing

- Optimized feature selection for J48 compatibility

8. Conclusions and Recommendations

8.1 Project Success Metrics

The sentiment analysis project successfully achieved its primary objectives:

- Data Collection: Exceeded target with 1,100+ unique reviews collected

- Model Development: Implemented functional J48 classification system

- Performance Achievement: Achieved 95.9% training accuracy and 87.0% test accuracy

- Technical Implementation: Successfully deployed WEKA-based solution

8.2 Recommendations for Future Enhancement

1. **Data Augmentation:** Collect more balanced datasets with equal positive and negative examples

2. **Advanced Preprocessing:** Implement more sophisticated text preprocessing including stemming and lemmatization

3. **Ensemble Methods:** Explore Random Forest or other ensemble approaches for improved generalization

4. **Feature Engineering:** Develop domain-specific features beyond basic bag-of-words representation

5. **Validation Strategy:** Implement stratified sampling to ensure balanced class distribution in test sets

8.3 Business Applications

This sentiment analysis system provides immediate value for:

- Product Quality Monitoring: Automated identification of customer satisfaction trends

- Marketing Intelligence: Real-time sentiment tracking for brand reputation management

- Customer Service Optimization: Priority flagging of negative feedback for immediate response

- Product Development: Data-driven insights for product improvement initiatives

9. Technical Specifications

9.1 System Requirements

- Platform: WEKA 3.8+ with Java 8+ runtime environment

- Data Format: ARFF (Attribute-Relation File Format)

- Algorithm: J48 Decision Tree (WEKA implementation of C4.5)

- Preprocessing: StringToWordVector filter for text transformation

9.2 Performance Benchmarks

- Model Training Time: < 1 second for 1,000 instances
- Prediction Time: < 0.1 seconds for 100 test instances
- Memory Requirements: Minimal (suitable for standard desktop deployment)
- Scalability: Linear scaling with dataset size

This comprehensive sentiment analysis project demonstrates the effective application of machine learning techniques to real-world text classification challenges, providing a solid foundation for automated customer feedback analysis systems.