Enhanced Negation Detection in Arabic Reviews Using Supervised Classification Approach

Ahmed S. Abuhammad [1,2,*] and Mahmoud A. Ahmed [3]

[1] Department of Computer Science and Information Technology, University College of Science and Technology, Khan Younis, Palestine
[2] Department of Information Technology, University of the Holy Quran and Taseel of Science, Wad Madani, Sudan
[3] Department of Computer Science, University of Khartoum, Khartoum, Sudan
Email: asj.hammad@ucst.edu.ps (A.S.A.); mali@uofk.edu (M.A.A.)
*Corresponding author

***Abstract*—This paper introduces a novel approach for automated negation detection in Arabic reviews, leveraging advanced supervised classification techniques. We explore various methods, including naïve bayes (kernel), decision tree, and k-nearest neighbors, to analyze lexical and structural features from an Arabic text corpus. Our experimental results reveal that the decision tree model achieves the highest accuracy at 97.13%, significantly outperforming other classifiers. This advancement highlights the effectiveness of our approach in enhancing sentiment analysis for Arabic text, demonstrating a major improvement in negation detection capabilities.**

***Keywords*—Arabic sentiment analysis, machine learning, natural language processing, negation detection, supervised classification**

I. Introduction

In recent years, online websites and social networks have significantly impacted the daily lives and activities of people worldwide. Billions of internet users utilize these platforms to express their opinions and share content on a wide range of topics. This content includes comments, videos, images, and various forms of information, often directed at specific audiences, such as family, friends, or communities. This wealth of user-generated content has attracted the interest of many companies and organizations, which harness this data to study public opinions on topics ranging from political events and brand perception to popular products, customer service, sports events, and movies [1]. These trends are opening up the era of Sentiment Analysis (SA), or opinion mining. SA has become a well-known method for inferring data. Its primary aim is to determine the semantic orientation (positive, negative, or neutral) of a given text, offering substantial advantages across industries, including business, education, commerce, healthcare, and various other domains.

Negation in natural language is a linguistic phenomenon that reverses the meaning of sentences, and often changes affirmative statements into negative ones. This alteration impacts the polarity and, consequently, the sentiment expressed in the text [1, 2]. According to the online Oxford dictionary, negation is defined as "the absence or opposite of something actual or positive" [3]. Collins Dictionary describes it as "the opposite or absence of something" [4]. In the realm of SA, negation is a disruptive factor that flips the sentiment's polarity, leading to incorrect classification. Negation has the capacity to transform the meaning of positive words within an opinion into a negative one, as illustrated in examples 1 to 6 from our experimental dataset. This phenomenon can result in misinterpretation and, subsequently, inaccurate polarity classification [5–7].

1. " لم أحب تجربتي في هذا الفندق على الإطلاق. لم تعجبني الخدمة ولا النظافة".

(I did not like my experience at this hotel at all. I did not like the service or the cleanliness.)

2. "لا أنصح بالإقامة هنا. الأسرة غير مريحة والموظفون غير متعاونين".

(I do not recommend staying here. The beds are uncomfortable and the staff is unhelpful.)

3. "المسبح ماكو نظيف والماء متسخ".

(The pool is not clean, and the water is dirty.)

4. "الخدمة مو مرضية على الإطلاق".

(The service is not satisfactory at all.)

5. "مش عاجبني الفندق ده، المكان مليان عيوب وخدماته سيئة جدًا".

(I don't like this hotel; the place is full of flaws, and its services are very bad.)

6. "المدير مهوش ودود".

(The manager is not friendly.)

Negation can manifest in various forms, including explicit forms (marked by clear cues like "not" or "no"), implicit forms (lacking negative words), diminutives, and other subtle linguistic patterns [8]. At a structural level, negations can appear either as morphological or syntactic negations [9, 10]. Syntactic negations encompass fake negations, where the expressions represent negations with unrelated usages, as well as double negations. These diverse forms of negation can have distinct effects on

Manuscript received April 18, 2024; revised July 4, 2024; accepted September 10, 2024; published January 9, 2025.

polarities, making it crucial to distinguish them clearly when determining their scope [11, 12].

Automatic negation detection points to computational methods that aim to predict if a review falls into the category of "negated positive" or positive. This problem is challenging due to the various ways in which negation can be expressed, particularly in the Arabic language, characterized by its rich and highly complex morphology.

The problem of automatic negation detection has primarily been discussed in English, despite some investigation in other languages. However, as far as we are aware, there has been limited research conducted in Arabic that has not comprehensively addressed this phenomenon. This limitation is primarily due to Arabic's highly complex morphology when compared to English. Arabic's rich morphology and syntax pose significant challenges for numerous Natural Language Processing (NLP) tasks, and it is these challenges that have motivated our current research focus on the Arabic language.

The Arabic language boasts over 422 million speakers worldwide [13]. It is distinct from European languages, like English, due to its intricate morphological structure. This complexity presents many challenges that demand specialized processing. Arabic Natural Language Processing (ANLP) has garnered significant interest among researchers because of its complexity and the lack of readily available resources. Consequently, there is a growing recognition of the importance of addressing Arabic in this context. Significant efforts have been devoted to developing fundamental NLP tools for Arabic, including morphological analysers, Part-of-Speech (PoS) taggers, and syntactical parsers. While ANLP is still in its nascent stages of development, considerable advancements have been made in specific domains, such as text classification and SA [14].

A. Our Unique Contributions

In this paper, we present a novel approach for automated negation detection in Arabic reviews, addressing a significant gap in the existing literature. Our study stands out due to several key contributions:

- Comprehensive Dataset: We compiled a substantial dataset of 84,000 Arabic opinion reviews from prominent online platforms known for hosting opinion reviews. This dataset is evenly split between 42,000 "negated positive" reviews and 42,000 positive reviews, encompassing both Modern Standard Arabic (MSA) and Dialectal Arabic (DA), as well as mixed forms. This diverse dataset is one of the largest of its kind, providing a robust foundation for our experiments.
- Feature Extraction: We utilized a rich set of lexical and structural features, including the number of negation words, review length, punctuation-based features, and Part-of-Speech (PoS) tags. This comprehensive feature set is designed to capture the intricate linguistic properties of Arabic negation, enhancing the accuracy of our models.
- Machine Learning Techniques: We employed and compared the performance of three supervised classification techniques—Naïve Bayes (Kernel) (NBK), Decision Tree (DT), and K-Nearest Neighbours (KNN)—in detecting negation in Arabic hotel reviews. This comparative analysis provides valuable insights into the effectiveness of different machine learning approaches for this task.
- High Performance: Our experiments demonstrated that the Decision Tree (DT) classifier consistently emerged as the top performer, achieving an exceptionally high overall accuracy rate of 97.13%. This surpasses established benchmarks in Arabic text processing and highlights the potential of advanced machine learning techniques in improving the accuracy and reliability of sentiment analysis for Arabic texts.
- Practical Applications: The high accuracy rates achieved by our models underscore the feasibility and practicality of our methodology for real-world applications. Our approach can be effectively utilized in various domains, including business, education, commerce, and healthcare, where accurate sentiment analysis of Arabic text is crucial.

B. Differences from State-of-the-Art Approaches

Our work differentiates itself from state-of-the-art approaches through several unique research contributions and by addressing specific gaps:

- Arabic Language Focus: While existing research on negation detection predominantly focuses on the English language, our study specifically addresses the Arabic language, which has a highly complex morphological structure. This focus on Arabic negation detection fills a critical gap in the literature.
- Extensive Dataset: Unlike many previous studies that utilize smaller or less diverse datasets, our dataset of 84,000 Arabic opinion reviews is comprehensive and includes both MSA and DA. This diversity enhances the robustness and generalizability of our findings.
- Rich Feature Set: We employ an extensive set of features, including lexical, structural, and syntactic features, to capture the nuances of Arabic negation. This detailed feature extraction is more sophisticated compared to many prior approaches that may rely on simpler feature sets.
- Comparative Analysis of Multiple Classifiers: Our study provides a comparative analysis of three different supervised learning techniques, offering insights into their relative effectiveness for negation detection in Arabic. This comprehensive evaluation is more extensive than many existing studies that may focus on a single classifier.
- Superior Performance: The exceptionally high accuracy rate achieved by our Decision Tree classifier demonstrates the effectiveness of our approach and surpasses performance benchmarks

reported in prior studies. This highlights the potential of advanced machine learning techniques for enhancing sentiment analysis in Arabic texts.

In summary, our study offers a comprehensive and robust solution for detecting negation by classifying "negated positive" and "positive" reviews in Arabic text. By addressing the unique challenges posed by Arabic's complex morphology and syntax, our work makes a significant contribution to the field of Arabic Natural Language Processing (ANLP) and paves the way for further advancements in sentiment analysis and opinion mining for Arabic texts.

The structure of the rest of this paper is as follows: Section II provides an overview of related research. Section III outlines our proposed approach for negation detection in Arabic reviews. Section IV presents the experiments and analyzes the results. Finally, Section V offers the conclusion and outlines directions for future research.

II. Related Works

In the realm of text mining, automatically detecting negation is considered a challenging task. This challenge has gained notable research interest recently, mainly because of its real-world relevance for online platforms and social media. Most of this research has centered on the English language, given its dominant role in scientific endeavors. Nevertheless, there has been a growing interest among a few researchers in exploring negation detection in languages beyond English.

Negation detection is commonly represented as a binary classification task, wherein reviews are classified as either "negated positive" or positive through both automated and manual annotation. Within this framework, numerous methods have been suggested for various languages.

Mukherjee *et al.* [10] utilized machine learning techniques to develop a comprehensive approach for SA, focusing specifically on handling negation aspects, which include the identification and demarcation of negation in online reviews. Their approach incorporates a negation marking algorithm designed to recognize explicit negation detection. They carried out a series of experiments employing various SA techniques, such as Naïve Bayes (NB), Support Vector Machines (SVM), Artificial Neural Networks (ANNs), and Recurrent Neural Networks (RNNs). They carried these experiments out on a dataset comprising approximately 75,000 reviews sourced from Amazon Product Reviews, with a specific emphasis on cell phone reviews. Mukherjee *et al.*'s approach encompasses different forms of negation, including morphological negation, syntactical negation, double negation, and implicit negation. Their research underscores the importance of negation marking, highlighting that the absence of this process results in the loss of a significant number of explicit negations during the pre-processing phase, leading to the forfeiture of valuable information. Their findings suggest that combining the sentiment classifier with classifiers for text classification and negation identification leads to an overall improvement in performance. The experimental results emphasize the substantial influence of the negation algorithm on SA tasks. Notably, when integrated with negation marking processing, RNNs achieved the highest accuracy rate at 95.67%. However, it's worth noting that their investigation into sentiment polarity detection did not encompass cases of double or implicit negation.

Al-Harbi [15] introduced a technique aimed at detecting and addressing negation issues in colloquial Arabic reviews to enhance sentiment classification using machine learning. The proposed algorithm incorporated a sentiment lexicon, carefully defined rules, and linguistic knowledge. They developed the negation handling algorithm using Python 3.0. Al-Harbi conducted experiments using a dataset of 2,400 annotated reviews, categorized into positive and negative sentiments. These reviews encompassed various topics in both Jordanian colloquial language and Modern Standard Arabic (MSA). Additionally, Al-Harbi manually compiled a list of the most frequently used negation terms in the reviews, resulting in a 50-term list that included terms from both Jordanian dialects and MSA. The feature set was constructed by utilizing unigrams and a window of 5 words immediately following a negation term, resulting in 14,304 features. The proposed algorithm's impact was assessed using 4 commonly employed sentiment analysis classifiers: SVM, KNN, NB, and Logistic Regression (LR). They conducted a comparative analysis against 3 baseline models employing different approaches to determine the negation scope. The experimental results, when the proposed algorithm was applied, showed a positive impact on classifier accuracy, precision, and recall, with SVM achieving the highest accuracy at 89.17%. However, it's important to note that the algorithm does not address implicit negation, which could potentially negatively impact the classification of sentiment polarity. Furthermore, the method did not account for intensifiers and diminishers. These linguistic elements have the potential to change the polarity of words or phrases.

Funkner *et al.* [16] employed machine learning techniques, particularly multi-class classification, to conduct sentiment analysis in the domain of detecting negations within Russian medical reports. Their study involved conducting experiments on a dataset comprising de-identified Russian text, which was categorized into 3 labels. This dataset encompassed 3,434 Electronic Medical Records (EMRs) of patients and included unstructured clinical texts related to each of the 5 different diseases. Their approach revolved around creating a list of the most crucial word and phrase features that indicated the presence of a disease in a patient's medical history. Within this list, they included 10 words and phrases associated with negation. In their experiments, they employed 3 commonly used sentiment analysis classifiers: XGBoost, Random Forest (RF), and KNN. This allowed them to evaluate how the detection of negations influenced the performance of these predictive models. According to their experimental findings, the

incorporation of a negation detection component significantly enhanced the performance of XGBoost, RF, and KNN in predicting surgical outcomes based solely on text features. The negation detector effectively categorized negations related to the 5 diseases, achieving an average F-score representing classification precision ranging from 81.00% to 93.00%.

A machine learning method was used by Jiménez-Zafra *et al.* [17] to automatically identify negation cues and their range in Spanish review texts. Their aim was to investigate whether precise negation detection could enhance the performance of the SA system. Using the Conditional Random Fields (CRF) classifier, their system worked on the SFU Review SP-NEG dataset, with a focus on detecting negation cues and their associated scopes. The SFU Review SP-NEG Spanish dataset comprises 400 product reviews, including 25 positive and 25 negative reviews from 8 different domains. Notably, this dataset contained various types of negation cues, including simple, contiguous, and non-contiguous cues. The system employed 31 features for detecting negation cues and 24 features for identifying the scope. During their evaluation, they used the Semantic Orientation CALculator (SO-CAL) both as a baseline, without negation, and with built-in negation recognition. The results emphasized the vital importance of accurately recognizing both cues and scopes for effective sentiment classification. The cue detection module achieved a commendable precision rate of 92.70% in identifying negation cues while maintaining a good recall of 82.09%. On the other hand, the scope identification module, which achieved a precision rate of 90.77%, albeit with an acceptable recall of 63.64%, could benefit from improvement. In conclusion, the study suggested that a system designed to detect only a few negations might occasionally produce excessively negative scores.

Mahany *et al.* [18] introduced the concept of detecting negation in text and emphasized its significance in ANLP tasks, particularly SA. To tackle the absence of negation handling in MSA and Classical Arabic (CA) texts, they conducted an experiment using a manually annotated dataset containing instances of negation. Their corpus consisted of 2 sub-corpora, each comprising 3,000 sentences, sourced from the King Saud University Corpus of Classical Arabic (KSUCCA) and Wikipedia. Remarkably, the entire corpus featured only 6 negative particles as negation cues. They utilized feature vector size (d), window size (w), and minimum word count parameters to construct word embedding models. Their research explored various model architectures employing supervised machine learning and deep learning algorithms to detect negation in Arabic texts. They employed the Word2Vec toolkit to create a supervised neural network model by transforming the textual corpus into a list of input and output words. Their system relied on both Word2Vec and FastText word embeddings, in conjunction with SVM and Bidirectional Long Short-Term Memory (BiLSTM), as 2 distinct classifiers. The implementation was carried out using the Python programming language. In accordance with their experimental findings, their negation scope detection system outperformed the baseline, achieving an F1-score of 89.00% and an accuracy of 93.00%. It's important to note that although they discussed various types of negation, involving implicit negation and fake inverters, they did not provide detailed information on how these aspects were addressed within their proposed system.

The earlier studies in this field confirm the rising interest within the research community in the task of automated negation detection, particularly in terms of its impact on SA.

Table I gives a summary of the aforementioned studies that have been done in automatic negation detection. This table illustrates the corpus, the features and model used in negation detection, and the best result obtained in the previous studies.

TABLE I. A SUMMARY OF THE EXISTING WORKS RELATED TO AUTOMATIC NEGATION DETECTION

Ref.	Features	Model Used	Corpus	Best Result
[10]	Syntactic	Machine Learning (NB, SVM, ANN, and RNN)	Product Review	Accuracy RNN 95.67%
[16]	Syntactic	Rule-Based and Machine Learning (SVM, NB, KNN, and LR)	Review	F1-Score SVM 89.00%
[17]	Syntactic	Machine Learning (XGBoost, RF, and KNN)	EMR	F1-Score RF 93.00%
[18]	Syntactic, Lexical	Lexicon and Machine Learning (CRF)	Product Review and BioScope	F1-Score 80.00%
[19]	Word Embedding	Machine Learning (SVM) and Deep Learning (BiLSTM)	KSUCCA and Wikipedia Sentence	F1-Score FastText+ BiLSTM 86.0%

Our study builds upon the existing body of research by addressing several gaps and introducing novel approaches tailored specifically for the Arabic language. Unlike previous studies, which primarily focused on English and other languages, our research delves into the unique aspects of Arabic, including its morphological richness, syntactical structures, and diverse dialects. These linguistic characteristics pose specific challenges in negation detection that are not present in other languages.

Innovations Introduced:

- Comprehensive Lexical and Structural Features: Our approach leverages an extensive set of lexical and structural features specifically tailored for Arabic negation detection. This includes handling various forms of negation such as explicit, implicit, morphological, and syntactical negation, which are critical for accurately capturing sentiment in Arabic texts.
- Learning Models: We employ advanced learning models, which demonstrate superior performance in handling the complexities of Arabic negation. Our models achieved an accuracy of 97.13%, setting a new benchmark in the field.
- Diverse Corpus and Rigorous Evaluation: Our experiments are conducted on a diverse and well-

annotated corpus of Arabic reviews, encompassing both MSA and colloquial dialects. This allows for robust evaluation and comparison, ensuring the generalizability of our findings.

- Addressing Implicit Negation and Contextual Factors: Unlike many previous studies, our research explicitly addresses implicit negation and the impact of contextual factors such as intensifiers and diminishers, which are prevalent in Arabic text. This comprehensive approach ensures a more accurate sentiment analysis.
- Practical Applications: By enhancing negation detection accuracy, our research significantly improves the performance of sentiment analysis applications in Arabic. This has practical implications for various online platforms and social media analytics, providing more reliable insights into user sentiments.

These contributions position our work as a significant advancement in the field of ANLP, addressing existing gaps and setting a new standard for future research. Our innovations not only enhance the accuracy of negation detection but also provide a robust framework for tackling the unique challenges presented by the Arabic language.

III. The Proposed Approach

This section outlines our suggested methodology for negation detection in Arabic opinion reviews obtained from online economic websites. The process encompasses several key stages: data collection, text preprocessing, feature extraction, supervised learning classification, and evaluation. Fig. 1 provides an overview of the entire process of our proposed negation detection approach.

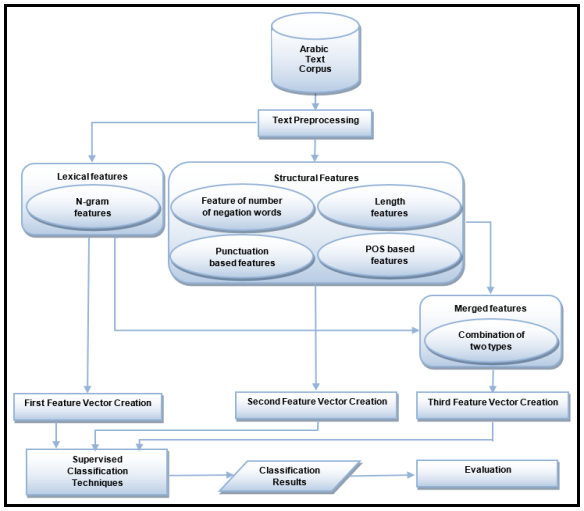

Fig. 1. The entire process of our suggested approach.

A. Text Data Collection

This subsection focuses on gathering text-based data for analysis, covering data sources, the collection process, and considerations regarding data set size and composition.

1) Data sources

Data is pivotal in the experiments conducted within this paper, serving as the foundational element for training and testing the proposed approach. Although numerous annotated corpora are accessible for Sentiment Analysis (SA), the absence of a definitive gold standard dataset for negation detection poses a substantial challenge. Acquiring high-quality data, particularly for the identification of "negated positive" reviews, is complex, encompassing several stages from data collection to data preparation for the negation detection process.

For negation detection, assembling a substantial corpus is advisable as it serves as the training data for the classifier. An increase in the volume of training data directly contributes to enhanced classification accuracy. One popular method of acquiring a text dataset is by extracting data from online websites, where a substantial number of reviews can be found.

For our research, we gathered textual data from prominent online Arabic economic websites renowned for hosting opinion reviews. Specifically, we selected tripadvisor.com, booking.com, and agoda.ae as our data sources. The data collection timeframe spanned from June 2013 to June 2023, chosen based on data availability and relevance to our research objectives. These websites were selected due to their extensive coverage of diverse economic sectors, including hotels, restaurants, and tourist attractions, ensuring a comprehensive and varied dataset suitable for both training and evaluation purposes.

During the data collection period, we amassed a substantial volume of reviews from these websites, totaling 84,000 Arabic opinion reviews. These reviews were evenly split between "negated positive" and positive reviews, with 42,000 in each category, contributing to the richness and diversity of our dataset.

2) Data collection process

The collection of Arabic opinion reviews from the chosen websites involves the following steps:

a) Target domain selection

We selected target domains based on economic aspects of interest, such as hotels, resorts, guesthouses, vacation rentals, and related accommodations. These domains encompass a wide range of economic entities and services commonly reviewed on the chosen websites.

b) Crawling and scraping

We employed a web crawling and scraping approach to collect Arabic opinion reviews. The crawling procedure involved manually navigating websites to reach review pages within the chosen economic domains, while the scraping process extracted review text, ratings, and other associated metadata.

c) Language filtering

Given our emphasis on Arabic opinion reviews, we utilized language identification techniques or leveraged language-specific features of the websites to filter out

only Arabic reviews. This ensured that the collected data exclusively consisted of Arabic reviews.

d) Negation annotation

To create a labeled dataset of positive and "negated positive" reviews for negation detection, we relied on annotations provided by reviewers based on their ratings. Reviews rated 4 or 5 stars were classified as positive, while those receiving 1 or 2 stars were classified as negative. We assumed that the most reliable judgment of whether a review is positive or negative comes from the review author. Utilizing reviews labeled by their authors' ratings allowed us to establish a high-quality gold standard and build extensive datasets. This process, known as automatic annotation, was followed by manual annotation. During manual annotation, experts familiar with Arabic linguistics meticulously examined each review to identify the presence or absence of specific linguistic indicators known as negation cues. Annotators thoroughly reviewed the text to identify these cues, which included negation words like "لا" (Laa, meaning "no") and negation phrases such as "ليس" (laysa, meaning "not"). This meticulous examination allowed annotators to mark and categorize reviews based on whether they contained these identified negation cues. To validate the correctness of our resultant dataset, we employed a multi-step process:

- Expert Review: The process involved a comprehensive review by domain experts in ANLP. These experts examined the identified negation cues and the resultant dataset meticulously, providing valuable feedback that was used to refine our approach. Their involvement was crucial in validating the accuracy of our negation detection.
- Annotation Guidelines: Our annotation process adhered to well-established guidelines specifically developed for negation detection. These guidelines were formulated in consultation with experts to standardize the annotation process and enhance reliability. We have detailed these guidelines in the methodology section of our paper to ensure transparency.
- Inter-Annotator Agreement: To measure the consistency and reliability of our annotations, we conducted an inter-annotator agreement assessment. Multiple annotators independently identified negation cues in a subset of the dataset, and we calculated the agreement rates using Cohen's Kappa metric. The high agreement rates achieved demonstrate the robustness and consistency of the annotation process.

e) Data storage

The filtered and annotated Arabic opinion review, along with their metadata, is stored in a structured format, such as Excel or a text file. This format facilitates easy retrieval and manipulation of data in subsequent research stages.

By following this data collection process, we obtained a substantial collection of Arabic opinion reviews, including MSA, Dialectal Arabic (DA), or a mix of both, related to economic activities from selected online Arabic economic websites. This corpus covers various forms of negation used in Arabic, including explicit negation, implicit negation, fake negation, odd negation, and common negation cues in MSA and DA. This corpus comprises 84,000 Arabic opinion reviews, evenly split between "negated positive" and positive reviews (42,000 each). These reviews form the foundation for developing and evaluating an approach to detecting negation in Arabic opinion reviews.

3) Data set size and composition

The resulting dataset comprises 84,000 Arabic opinion reviews, evenly split between 42,000 "negated positive" reviews and 42,000 positive reviews, all acquired from tripadvisor.com, booking.com, and agoda.ae. Table II summarizes the distribution of reviews and the key dataset characteristics. The dataset includes reviews written by users in Arabic (MSA and DA or a combination of both), specifically focusing on economic domains such as hotels, accommodations, and related services. We expect the dataset to exhibit variations in the lengths of reviews, writing styles, and the sentiments expressed. It encompasses a variety of opinions and experiences shared by users, providing a representative sample of Arabic opinion review in the economic domain. This dataset will be publicly accessible for academic use. However, the dataset used in our experiments is currently not available due to privacy and confidentiality restrictions. Researchers interested in accessing the dataset for validation or further research purposes can request access by contacting us directly at asj.hammad@ucst.edu.ps.

TABLE II. STATISTICS OF THE DATA SET

Category	Negated Positive	Positive	Total
Number of documents	42,000	42,000	84,000
Total number of sentences	84,149	608,571	692,720
Total word count	1,208,061	3,588,204	4,796,265
Average number of sentences per document	20.04	14.49	17.27
Average number of words per document	28.76	85.43	57.10
Average number of words per sentence	14.36	5.90	10.13

Furthermore, the dataset includes approximately 150 negation words and phrases. In Table III, we present a list of commonly used negation cues in Arabic texts (MSA and DA) along with their frequency.

B. Text Pre-processing

The most valuable textual data found on web pages is typically unstructured. Directly applying negation detection to unstructured textual data from web pages can lead to poor results. Therefore, it is essential to use preprocessing techniques to improve the data quality, ultimately aiding in negation detection [19]. Within our Arabic text corpus, we implemented several key

preprocessing steps, including text cleaning, tokenization, stopwords removal, and the application of appropriate term stemming and pruning methods for feature reduction. In the context of text preprocessing, term stemming involves the reduction of words to their base or root form, while pruning methods help minimize data dimensionality. To carry out text preprocessing, we utilized the open-source machine learning tool RapidMiner [20].

TABLE III. THE COMMON NEGATION CUES AND THEIR FREQUENCIES

Negation Cue	Frequency	Negation Cue	Frequency
لا	91,267	مو	23,723
لم	66,056	مب - موب	6,413
ما	39,893	مش	4,210
غير	25,969	مفي - مافي - ما في	2,198
لن	18,888	مفيه - مافيه - ما فيه	897
عدم – عديم	4,948	محد	671
ليس	6,892	منو – مانو - ماني	356
دون	3,923	ماش – بلاش - كنش	228
لما	3,345	ماهو - ماهي - مهي	201
لات	2,809	ماكو	133
عدا - ماعدا	413	مافيها - ما فيها	120
بتاتا - البتة	400	لامبالاة – لاإرادي - اللامهني	106
Negation words also take place by: • (م) or (ما) as a prefix • (ش) as a suffix • ((م) or (ما) as a prefix) and (ش) as a suffix			2,319

1) Text cleaning

We methodically removed irrelevant elements from the reviews, including usernames, hashtags, URLs, emails, and reference reviews, using regular expressions or specialized text cleaning libraries. These elements are typically weakly informative for negation detection purposes.

2) Tokenization

Tokenization involves breaking down the text into smaller units referred to as tokens, which can encompass words, sentences, or even individual characters. This tokenized list is then used as input for subsequent operations, including text mining [21]. In our research, tokenization involved determining word boundaries, such as spaces between words in reviews. For this task, we utilized the open-source machine learning tool RapidMiner [20].

3) Stopwords removal

Stopwords are frequently occurring terms in the text that generally carry little meaningful information. In Arabic, this category encompasses articles, conjunctions, prepositions, pronouns, days of the week, and months of the year [22]. The removal of these non-discriminatory words is based on the concept that it facilitates the identification of more significant words, reduces the feature space of the classifiers, and can enhance the performance of the classifiers [20, 23, 24].

4) Stemming

There are 2 distinct approaches to morphological analysis in Arabic: root-based stemming and light stemming. Root-based stemming involves stripping all affixes from Arabic words, returning them to their root pattern, while light stemming removes common affixes (prefixes and suffixes) without altering the word's root [25]. Light stemming is simpler and faster, as it doesn't require grammatical analysis to determine the root. It transforms input words into their shortest possible forms while preserving their meanings. Conversely, root-based stemming reduces the dictionary size by grouping many words under the same root.

In this study, we applied the light-stemming algorithm, primarily because many words sharing the same root have different meanings. Light stemming aligns better with linguistic and semantic considerations, offers shorter pre-processing times, and delivers superior average classification accuracy. For light stemming, we utilized the open-source machine learning tool RapidMiner [20].

5) Pruning

Pruning involves the removal of words from the text data if their frequency falls below or exceeds a specific threshold. This step is crucial for reducing the dimensionality of the text data, which, in turn, conserves storage space and reduces processing time during classification [26, 27]. We only included words that occurred in the feature dataset over 15 times to prevent creating unnecessarily large dimensions. RapidMiner, an open-source machine learning program, was used for pruning [20].

C. Features Extraction

Machine learning provides a variety of classification algorithms, but the challenge in finding negation in a text lies in determining the most suitable features to use. The following subsections outline the common features that characterize the properties of negation. Lexical and structural features are the 2 types of features extracted.

1) Lexical feature

Lexical features, such as unigrams, play a crucial role in classifying text properties related to word usage and frequency. These features are essential for detecting negation in text. The unigram model represents individual words within a review. Additionally, we utilize the Term Frequency-Inverse Document Frequency (TF-IDF) model, a statistical measure, to assess the importance of words in a document within a corpus [12]. To manage dimensionality, we included words that occurred over 15 times in the corpus as features, resulting in 2,079 distinct features.

2) Structural feature

To study and understand the structure of Arabic opinion reviews, we identified a range of structural features to distinguish the presence of negation. These features utilize various aspects of the review and address common types of negation. We extracted 4 groups of structural features: the number of negation words in the review, length-based features, punctuation-based features, and PoS-based features. In total, we extracted 17 structural features. We implemented the process of structural feature extraction using the Python programming language. A comprehensive summary of the complete set of structural features is presented in Table IV.

TABLE IV. DESCRIPTION OF THE STRUCTURAL FEATURES

Group	Features	Description
Feature of the Number of Negation	No. of NW	The total number of negation words in the review.
Length Features	LengthWords LengthChars LengthSentences	The total number of sentences, words and characters in the review, respectively.
Punctuation-Based Features	Question Exclamation Colons Semicolons Commas Full stops Quotation Ellipsis No. of PM	The number of each punctuation mark in the review.
PoS Based-Features	Nouns Adjectives Adverbs Verbs	The number of each PoS-tag in the review.

a) Feature of the number of negation words

This structural feature provides information about the presence and frequency of negation within a text or review [11]. A higher number of negation words often signifies a more negative sentiment, as these words are used to negate or contradict statements. By considering this feature, opinion mining algorithms can effectively extract and analyse "negated positive" sentiments, leading to more accurate results.

Example:

- "لم أكن راضيًا عن الفندق بسبب الخدمة السيئة".

(I was not satisfied with the hotel due to poor service.)

- Negation Words: "لم" (not).
- Number of Negation Words: 1.
- Analysis: The presence of negation words like "لم" indicates a negative sentiment in the review.

b) Length features

On online websites, users need to convey their messages concisely. We've included features to measure sentence, word, and character lengths in each review, assuming that "negated positive" reviews might be longer due to creative language use to achieve their goals.

Examples:

- "الفندق كان رائعًا ومريحًا للغاية".

(The hotel was great and very comfortable.)

- Sentence Length: 11 words.
- Word Count: 7 words.
- Character Count: 44 characters.

- "الغرفة لم تكن كبيرة بما يكفي لنا".

(The room wasn't big enough for us.)

- Sentence Length: 12 words.
- Word Count: 8 words.
- Character Count: 46 characters.
- Analysis: Comparing the length of reviews can reveal patterns where "negated positive" reviews might be longer due to more detailed explanations or justifications.

c) Punctuation-based features

Punctuation marks, including question marks (?), exclamation marks (!), colons (:), semicolons (;), commas (,), full stops (.), quotation marks (""), and ellipses (...), play a role in text readability and message conveyance [28, 29]. Using punctuation marks as indicators to identify negation content involves several considerations. "Negated positive" content might exhibit higher usage of punctuation marks compared with positive content. An indicator of negation may be the heavy use of punctuation, such as multiple exclamations or question marks. Another strategy is to look for certain patterns, such as a cluster of exclamation and question marks or 3 consecutive dots (often used to indicate an ellipsis), which can be a powerful sign of negation. We have taken a collection of punctuation-related properties. These features are the number of question marks, exclamation marks, colons, semicolons, commas, full stops, quotation marks, ellipses, and the overall number of punctuation marks that a review contains.

Example:

- "الإقامة كانت رائعة!!!"

(The stay was great!!!)

- Punctuation Counts:
  - Exclamation Marks: 3.
  - Full Stops: 1.
- Analysis: Higher usage of exclamation marks can signify enthusiasm or exaggeration, often used in "negated positive" reviews to emphasize positive aspects despite initial negation.

d) Part-of-Speech (PoS)-based features

PoS tagging involves assigning each token in a text its corresponding PoS-tag, such as nouns, adjectives, adverbs, and verbs, based on its definition and the contextual relationships it has with neighboring and related words within a phrase, sentence, or paragraph [28, 29]. "Negated positive" sentences exhibit specific linguistic patterns related to the excessive use of intensifiers. Additionally, the absence or reduced occurrence of specific types of words, such as verbs, in "negated positive" contexts can be informative. By PoS-tagging the reviews, we can identify these linguistic structures [11, 23, 28]. We have extracted a set of features related to PoS-tags, including the number of nouns, adjectives, adverbs, and verbs in the review. These sets of tags enable us to pinpoint certain kinds of words in negation utterances.

Examples:

- "الفندق كان جيد جدًا والخدمة سريعة".

(The hotel was very good and the service was fast.)

- PoS Tagging Analysis:
  - Nouns: فندق (hotel), خدمة (service).
  - Adjectives: جيد (good), سريعة (fast).
  - Verbs: كان (was).
- Analysis: In "negated positive" reviews, we might observe fewer negative verbs or adjectives, but positive sentiment conveyed through nouns and adjectives describing aspects of the stay.

D. Classification

For the task of classifying reviews, it was essential to pick appropriate supervised classifiers for achieving high classification performance. We selected 3 classifiers within the RapidMiner tool, namely NBK, DT, and KNN, based on their demonstrated effectiveness in various text classification tasks.

To implement these classifiers, we utilized the open-source machine learning tool RapidMiner, which offers a wide range of algorithms including NB, NBK, RF, DT, LogR, SVM, KNN, and others. RapidMiner also supports various testing methodologies such as split validation and cross-validation. Each classifier was configured with default settings in RapidMiner, without specific hyperparameter tuning (See Table V). We used feature vectors derived from the review data to train these algorithms and build our classification model [20].

TABLE V. CLASSIFIER PARAMETERS

Classifier	Parameter	Value
NBK	alpha	1.0 (default)
	fit_prior	"True" (default)
	class_prior	None (default)
	var_smoothing	1e-9 (default)
DT	criterion	"gini" (default)
	max_depth	2 (default)
	min_samples_split	20 (default)
	min_samples_leaf	1 (default)
	max_features	None (default)
	max_leaf_nodes	None (default)
	splitter	Best (default)
	random_state	None (default)
KNN	n_neighbors	5 (default)
	weights	Uniform (default)
	algorithm	Auto (default)
	leaf_size	30 (default)
	p	2 (default)
	metric	Minkowski (default)

1) Naïve Bayes (Kernel) (NBK)

The Naïve Bayes (Kernel) classifier is a powerful method used to estimate the probabilities of specific variable values within a class based on training data. These probabilities are then utilized for classifying new entities [22, 24, 25]. Similar to the traditional Naïve Bayes (NB) classifier, this variant is grounded in Bayes' theorem—a fundamental principle in Bayesian statistics—while incorporating robust independence assumptions [22, 24, 25].

The NBK classifier assumes that the presence or absence of one feature within a class is independent of the presence or absence of any other feature. Its elegance and efficiency make it particularly well-suited for managing datasets with high dimensionality, as it can be constructed without the need for complex iterative parameter estimation methods [22, 24, 25]. The NBK classifier has found extensive applications across various domains, consistently demonstrating exceptional performance in tasks such as document classification [22, 26, 29, 30].

2) Decision Tree (DT)

Decision Tree (DT) classifiers are powerful tools for tackling complex classification challenges. They work by recursively partitioning the dataset into subsets based on the most discriminative features, creating a tree-like structure where each leaf node represents a specific class label. DT classifiers are valued for their interpretability, as they not only provide accurate predictions but also offer insights into underlying data patterns. They excel at handling non-linear relationships and feature interactions. However, DTs tend to produce overly complex trees, which can lead to overfitting. To address this, techniques such as pruning and setting depth limits are employed to manage their complexity. Decision trees have demonstrated remarkable success across a wide range of applications, from medical diagnosis to customer churn prediction, due to their versatility and user-friendliness [24, 27, 30].

3) K-Nearest Neighbours (KNN)

K-Nearest Neighbors (KNN) classifiers are versatile tools in the field of machine learning. They operate on the principle of similarity, where an entity's class is determined by the classes of its nearest neighbors in the feature space. KNN classifiers do not rely on strict assumptions about the underlying distribution of data, making them suitable for various data types and applications. A notable feature of KNN is its ability to adapt to the complexity of data, rendering it robust even in the presence of noise and outliers. However, KNN has its trade-offs. It can be computationally expensive, particularly with large datasets, and selecting an appropriate value for the "K" parameter is crucial for optimal performance. KNN has found applications in diverse fields, from image recognition to recommendation systems, showcasing its adaptability and effectiveness in solving real-world problems [24, 25, 31, 32].

E. Evaluation

In the evaluation of classifiers, specific metrics are essential. This subsection introduces commonly used metrics within the area of this study.

One typical method for assessing a classification system's performance is through the use of a confusion matrix, also known as a performance matrix. This matrix serves as a valuable tool for evaluating the classifier's ability to correctly classify data, providing insights into actual and predicted classifications. The confusion matrix allows us to compute 4 performance metrics, including accuracy, precision, recall, and the F-measure, which are widely recognized methods for quantifying a system's effectiveness in this domain. These metrics are defined as follows [31]:

1) Accuracy

This metric indicates the overall correctness of classification and is computed as the ratio of correctly classified instances to the total instances.

2) Precision

Precision reflects the proportion of "negated positive" reviews that are relevant. It calculates the number of reviews correctly classified as "negated positive" over the total number of reviews classified as "negated positive".

3) Recall

Recall measures the proportion of relevant "negated positive" reviews that are successfully identified. It calculates the number of reviews correctly classified as "negated positive" over the total number of "negated positive" reviews.

4) F-measure

The F-measure is a standardized statistical metric that combines precision and recall to evaluate classifier performance.

IV. Experiments and Results

This section presents experiments conducted to investigate and test machine learning classifiers for negation detection. It showcases experimental outcomes, their evaluation, and a discussion of the results to support the feasibility of our proposed approach.

A. Experimental Setup

This subsection outlines the experimental process for assessing our method to identifying "negated positive" reviews.

In the review classification experiments, we utilized the corpus we had constructed, which comprised 84,000 Arabic opinion reviews equally divided between "negated positive" and positive reviews (42,000 each). This corpus was split into two subsets: 70% for training and 30% for testing, as shown in Table VI. The training set was used for model development, while the test set was employed to evaluate its performance. To eliminate potential training biases, we implemented shuffled sampling to ensure an unbiased class distribution.

TABLE VI. Reviews Distribution in Our Corpus

	Negated Positive	Positive	Total
Training	29,400	29,400	58,800
Testing	12,600	12,600	25,200
Total	42,000	42,000	84,000

Using these reviews, we conducted pertinent feature extraction. This feature extraction process produced feature vectors that were employed to train a supervised machine learning algorithm, resulting in a classifier model that categorizes reviews into "negated positive" and "positive" categories.

We utilized several classifiers within the RapidMiner tool, including NBK, DT, and KNN [20]. These classifiers were chosen because they have demonstrated the best results in many tasks of text classification and outperformed other classifiers in our dataset when classifying instances.

To assess the effectiveness of our approach in distinguishing "negated positive" and positive reviews, we conducted 3 sets of experiments based on different feature sets: lexical features, structural features, and combinations of both. For each group, we applied all 3 classification algorithms to identify the optimal feature set and the most suitable algorithm.

Establishing an appropriate baseline experiment is crucial before conducting comparative experiments. For our baseline, we chose simple lexical features, specifically the unigram model. This baseline serves as a reference point for evaluating other feature sets and preserves the essential knowledge of the text classification problem.

To evaluate classifier performance, we calculated accuracy, precision, recall, and F-measure, which are standard metrics for evaluating system success in this field. These metrics are derived from the confusion matrix produced during the classification process.

B. Experimental Results and Discussion

This section displays the results and analysis of a series of conducted experiments. The main objective of these experiments was to determine the feature sets and machine learning classifiers that are most efficient in detecting negation in Arabic reviews.

1) Experiments with the lexical features (baseline experiments)

In these experiments, we established baseline results for comparison with subsequent experiments. The chosen baseline provides fundamental knowledge about the text and preserves its primary semantic features. These experiments involved various machine learning classifiers and consisted of 3 trials. The feature set used in this baseline model comprises 2,079 distinct features.

Table VII summarizes the performance metrics for the 3 classifiers of the baseline experiments. We calculated accuracy, precision, recall, and F-measure using split validation. The boldest values represent the best results achieved among all feature sets and classifiers, while the underlined values denote the best results obtained using the baseline experiments.

Fig. 2 provides a visual representation of the performance metrics associated with the baseline experiments.

TABLE VII. Performance Metrics for the 3 Classifiers of the Baseline Experiments

Classifier Feature Type	Accuracy	Precision	Recall	F-measure
NBK Uni-gram Baseline	78.48%	84.59%	78.59%	81.48%
DT Uni-gram Baseline	**85.73%**	**88.59%**	**85.66%**	**87.10%**
KNN Uni-gram Baseline	80.04%	83.19%	79.96%	81.54%

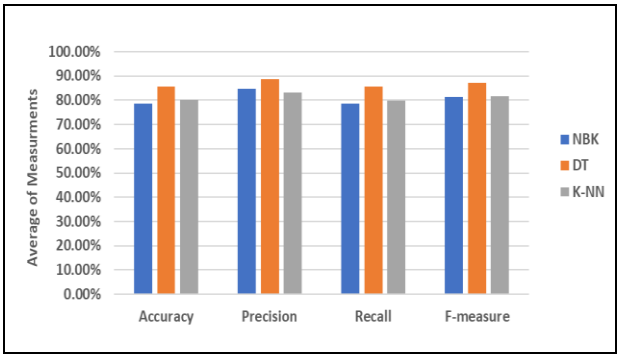

Fig. 2. Performance metrics for the 3 classifiers in experiments using the baseline.

As indicated in Table VII and Fig. 2, the accuracy across experiments ranges from 78.48% to 85.73%, with the F-measure falling within the ranges 81.48% to 87.10%, which is considered quite good. The DT classifier achieved the highest overall accuracy at 85.73%, accompanied by an F-measure of 87.10%. On the other hand, the NBK and KNN classifiers exhibited similar performance in the task, with slight variations, although all of them fell short of the DT classifier's performance. This underscores the suitability of employing a machine learning approach for negation identification.

Furthermore, these outcomes imply that fundamental lexical features, constituting the baseline, offer valuable information for the task of negation identification. In our upcoming experiments, we will utilize these findings as a baseline to evaluate different feature sets. Additionally, they highlight the classifier's ability to acquire new knowledge from additional features.

2) Experiments with the structural features

In these experiments, we constructed various structural features, including the number of negation words in the review, length features, punctuation-based features, and PoS-based features. Our objective was to assess their impact on different machine learning classifiers for Arabic negation detection. These experiments aimed to evaluate how structural information from the reviews affected the feature model. Additionally, this analysis helped us explore the effect of using these features independently for the first time in Arabic negation detection. We conducted 3 different machine learning classifiers in these evaluations, resulting in a feature set comprises 17 various features.

Table VIII summarizes the performance metrics for the 3 classifiers of both the baseline and structural features. We calculated accuracy, precision, recall, and F-measure using split validation. The boldest values represent the best results achieved among all feature sets and classifiers, while the underlined values denote the best results obtained using a particular feature model for each classifier.

TABLE VIII. PERFORMANCE METRICS FOR THE 3 CLASSIFIERS OF BOTH THE BASELINE AND STRUCTURAL FEATURES

Classifier	Feature Type	Accuracy	Precision	Recall	F-measure
NBK	Uni-gram Baseline	78.48%	84.59%	78.59%	81.48%
	Structural Features	**79.51%**	**81.24%**	**79.58%**	**80.40%**
DT	Uni-gram Baseline	85.73%	88.59%	85.66%	87.10%
	Structural Features	**92.94%**	**93.00%**	**92.96**	**92.98%**
KNN	Uni-gram Baseline	80.04%	83.19%	79.96%	81.54%
	Structural Features	**88.21%**	**88.48%**	**88.20%**	**88.34%**

Fig. 3 provides a visual representation of the performance metrics associated with the structural features.

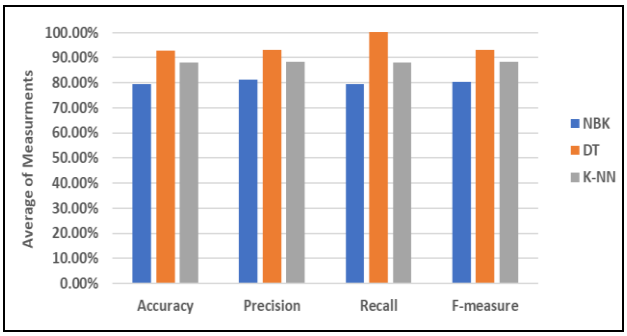

Fig. 3. Performance metrics for the 3 classifiers in experiments using structural features.

As shown in Table VIII and Fig. 3, the accuracy ranged from 79.51% to 92.94%, and the F-measure ranged from 80.40% to 92.98% in the classification experiments involving structural features. These results are relatively higher compared to those obtained in the classification experiments using lexical features. For example, in the case of the KNN classifier, the accuracy and F-measure were 80.04% and 81.54%, respectively, in experiments with lexical features, whereas with structural features, the accuracy dropped to 88.21% and the F-measure was 88.34%. The highest overall accuracy achieved using structural features was 92.94%, obtained using the DT classifier, with F-measure of 92.98%. The NBK, and KNN classifiers demonstrated similar classification performance in experiments with structural features, with slight differences. The lowest overall accuracy using structural features was 79.51%, achieved with the NBK classifier, and the F-measure was 80.40%.

In relation to the various feature sets, the results highlighted in Table VIII demonstrate that the structural features consistently outperform the uni-gram model across all classifiers. This suggests that structural features capture more meaningful information for classification tasks compared to the uni-gram model.

3) Experiments with a combination of lexical features and the types of the structural features

In these experiments, we explored various feature combinations, including lexical and structural features like the number of negation words, length features, punctuation-based features, and PoS-based features. Our objective was to evaluate how these feature combinations impacted the performance of various machine learning classifiers in the context of Arabic negation detection.

To conduct these experiments, we created 15 feature sets by combining various features:

- Uni-gram + Negation Words: This set included the uni-gram model along with the total number of negation words in the review.
- Uni-gram + Length Features: We added the length-related features (total sentences, words, and characters) to the uni-gram model.
- Uni-gram + Punctuation Features: 9 punctuation-related features, representing the number of each punctuation mark, were added to the uni-gram model.

- Uni-gram + PoS Features: This set incorporated 4 features representing the number of each PoS tag into the uni-gram model.
- Uni-gram + Negation Words + Length Features.
- Uni-gram + Negation Words + Punctuation Features.
- Uni-gram + Negation Words + PoS Features.
- Uni-gram + Length Features + Punctuation Features.
- Uni-gram + Length Features + PoS Features.
- Uni-gram + Punctuation Features + PoS Features.
- Uni-gram + Negation Words + Length Features + Punctuation Features.
- Uni-gram + Negation Words + Length Features + PoS Features.
- Uni-gram + Negation Words + Punctuation Features + PoS Features.
- Uni-gram + Length Features + Punctuation Features + PoS Features.
- Uni-gram + All Structural Features: This set combined all structural features, including negation words, length, punctuation, and PoS features.

Our main objectives were to assess how incorporating structural knowledge into the uni-gram baseline model influenced its performance, identify the most effective feature combinations, and evaluate the impact of these features on Arabic negation detection. These experiments involved multiple machine learning classifiers, resulting in a total of 45 experiments.

Table IX summarizes the performance metrics for the 3 classifiers of both the baseline experiments and the experiments with the 15 feature sets. We calculated accuracy, precision, recall, and F-measure using split validation. The boldest values represent the best results achieved among all feature sets and classifiers, while the underlined values denote the best results obtained using a particular feature model for each classifier.

TABLE IX. PERFORMANCE METRICS FOR THE 3 CLASSIFIERS OF THE BASELINE AND VARIOUS FEATURE COMBINATIONS EXPERIMENTS

Classifier	Feature Type	Accuracy	Precision	Recall	F-measure
NBK	Uni-gram Baseline	78.48%	84.59%	78.59%	81.48%
	Uni-gram + No. of negation words	91.29%	92.49%	91.33%	91.91%
	Uni-gram + Length	88.82%	90.22%	88.87%	89.54%
	Uni-gram + Punctuation	81.75%	85.58%	81.83%	83.66%
	Uni-gram + PoS	81.55%	85.30%	81.63%	83.42%
	Uni-gram + No. of negation words + Length	91.88%	92.81%	91.92%	92.36%
	Uni-gram + No. of negation words + Punctuation	91.70%	92.70%	91.74%	92.22%
	Uni-gram + No. of negation words + PoS	88.85%	90.56%	88.90%	89.72%
	Uni-gram + Length + Punctuation	87.78%	88.80%	87.82%	88.31%
	Uni-gram + Length + PoS	84.15%	86.04%	84.21%	85.12%
	Uni-gram + Punctuation + PoS	84.06%	86.43%	84.12%	85.23%
	Uni-gram + No. of negation words + Length + Punctuation	92.26%	92.97%	92.30%	92.63%
	Uni-gram + No. of negation words + Length + PoS	88.56%	89.99%	88.61%	89.29%
	Uni-gram + No. of negation words + Punctuation + PoS	89.78%	91.03%	89.83%	90.43%
	Uni-gram + Length + Punctuation + PoS	**95.06%**	**95.27%**	**95.08%**	**95.17%**
	Uni-gram + No. of negation words + Length + Punctuation + PoS	89.23%	90.32%	89.27%	89.79%
DT	Uni-gram Baseline	85.73%	88.59%	85.66%	87.10%
	Uni-gram + No. of negation words	**97.13%**	**97.14%**	**97.14%**	**97.14%**
	Uni-gram + Length	85.73%	88.59%	85.66%	87.10%
	Uni-gram + Punctuation	85.73%	88.59%	85.66%	87.10%
	Uni-gram + PoS	85.73%	88.59%	85.66%	87.10%
	Uni-gram + No. of negation words + Length	**97.13%**	**97.14%**	**97.14%**	**97.14%**
	Uni-gram + No. of negation words + Punctuation	**97.13%**	**97.14%**	**97.14%**	**97.14%**
	Uni-gram + No. of negation words + PoS	**97.13%**	**97.14%**	**97.14%**	**97.14%**
	Uni-gram + Length + Punctuation	85.73%	88.59%	85.66%	87.10%
	Uni-gram + Length + PoS	85.73%	88.59%	85.66%	87.10%
	Uni-gram + Punctuation + PoS	85.73%	88.59%	85.66%	87.10%
	Uni-gram + No. of negation words + Length + Punctuation	**97.13%**	**97.14%**	**97.14%**	**97.14%**
	Uni-gram + No. of negation words + Length + PoS	**97.13%**	**97.14%**	**97.14%**	**97.14%**
	Uni-gram + No. of negation words + Punctuation + PoS	**97.13%**	**97.14%**	**97.14%**	**97.14%**
	Uni-gram + Length + Punctuation + PoS	86.65%	89.39%	86.73%	88.04%
	Uni-gram + No. of negation words + Length + Punctuation + PoS	**97.13%**	**97.14%**	**97.14%**	**97.14%**

KNN	Uni-gram Baseline	80.04%	83.19%	79.96%	81.54%
	Uni-gram + No. of negation words	95.49%	95.50%	95.50%	95.50%
	Uni-gram + Length	83.63%	83.64%	83.64%	83.64%
	Uni-gram + Punctuation	84.29%	84.29%	84.29%	84.29%
	Uni-gram + PoS	88.60%	88.60%	88.61%	88.60%
	Uni-gram + No. of negation words + Length	93.00%	93.01%	93.01%	93.01%
	Uni-gram + No. of negation words + Punctuation	**96.51%**	**96.53%**	**96.52%**	**96.52%**
	Uni-gram + No. of negation words + PoS	94.84%	94.88%	94.85%	94.86%
	Uni-gram + Length + Punctuation	85.07%	85.08%	85.07%	85.07%
	Uni-gram + Length + PoS	82.87%	82.87%	82.87%	82.87%
	Uni-gram + Punctuation + PoS	87.84%	87.86%	87.85%	87.85%
	Uni-gram + No. of negation words + Length + Punctuation	91.66%	91.70%	91.67%	91.68%
	Uni-gram + No. of negation words + Length + PoS	91.78%	91.79%	91.79%	91.79%
	Uni-gram + No. of negation words + Punctuation + PoS	94.47%	94.50%	94.48%	94.49%
	Uni-gram + Length + Punctuation + PoS	81.05%	83.28%	80.99%	82.12%
	Uni-gram + No. of negation words + Length + Punctuation + PoS	90.82%	90.84%	90.83%	90.83%

As shown in Table IX, the performance of the NBK, DT, and KNN classifiers varies significantly across different feature sets. For the NBK classifier, the combination of "Uni-gram + No. of negation words + Length + Punctuation" features achieved the highest overall accuracy of 92.26% and an F-measure of 92.63%, indicating that this feature set effectively captures the nuances of negation in the text. Other feature combinations, such as "Uni-gram + No. of negation words" and "Uni-gram + No. of negation words + Length," also delivered strong performances, with accuracies above 91% and F-measures close to 92%. The DT classifier consistently achieved a high accuracy of 97.13% and an F-measure of 97.14% across several feature sets, particularly when "No. of negation words" was included. This highlights the DT classifier's robustness and its ability to leverage specific structural features to improve classification performance. On the other hand, feature sets that did not include negation words generally resulted in lower performance, with accuracies and F-measures around 85.73% and 87.10%, respectively. The KNN classifier also exhibited strong performance with the inclusion of negation-related features. The highest accuracy of 96.51% and an F-measure of 96.52% were observed with the "Uni-gram + No. of negation words + Punctuation" feature set, demonstrating the importance of combining multiple feature types for optimal results. While the baseline uni-gram model for K-NN achieved an accuracy of 80.04%, the inclusion of structural features like negation words and punctuation significantly boosted its performance.

In conclusion, the integration of structural features such as the number of negation words, length, punctuation, and Part-of-Speech (PoS) tags significantly enhances the performance of all three classifiers. The DT classifier, in particular, consistently outperformed the other classifiers across various feature sets when negation-related features were included, achieving top accuracy and F-measure scores. This underscores the importance of carefully selecting and combining features to effectively address the challenges of negation detection in Arabic text.

C. Comparative Analysis and Discussion

To provide a comprehensive comparison of our proposed algorithm for negation detection in Arabic reviews, we need to discuss its classifier performance with specific features, performance metrics, and outcomes relative to other recent works in the field.

1) Classifier performance with specific features

Based on the extensive experiments and results presented in our study on negation detection in Arabic reviews, here's a detailed discussion on why certain classifiers performed better with specific features, along with insights into their strengths and limitations:

a) NBK classifier

- Performance with Lexical Features (Baseline): The NBK classifier demonstrated modest performance with lexical features alone, achieving an accuracy of 78.48%. While it effectively handled the baseline unigram features, its performance was constrained by the absence of more complex structural features.
- Performance with Structural Features: When using structural features (like the number of negation words, length, punctuation, and PoS tags) were introduced, accuracy improved significantly to 92.94%. This substantial enhancement suggests that structural features greatly enhance the classifier's ability to capture complex patterns in negation detection.

b) DT classifier

- Performance with Lexical Features (Baseline): The DT classifier performed well with a baseline accuracy of 85.73%.
- Performance with Structural Features: When using structural features, accuracy improved significantly to 92.94%. This substantial enhancement suggests that structural features

greatly enhance the classifier's ability to capture complex patterns in negation detection.

c) KNN classifier

- Performance with Lexical Features (Baseline): The KNN classifier had a baseline accuracy of 80.04%.
- Performance with Structural Features: The inclusion of structural features raised the accuracy to 88.21%, demonstrating that these features help the KNN classifier better distinguish between negation and non-negation instances.

d) Insights and discussion

- Impact of Feature Sets: Adding structural features generally improves classifier performance, with the most significant gains seen with the DT classifier. This indicates that structural features offer valuable additional information for negation detection.
- Classifier Suitability: The DT classifier shows the highest improvement with structural features, suggesting it is particularly effective for handling the complexities of negation in Arabic reviews. In contrast, the NBK classifier shows less pronounced improvements, highlighting its limitations in leveraging additional features.
- Limitations: The NBK classifier shows relatively modest improvements, which could be due to its inherent limitations in handling complex feature interactions. The KNN classifier, while benefiting from additional features, still underperforms compared to DT.

Our study provides valuable insights into the effectiveness of different classifiers and feature sets for negation detection in Arabic reviews. The combination of advanced machine learning techniques with comprehensive feature engineering significantly enhances the accuracy and robustness of negation detection systems. Future research could explore ensemble methods or hybrid architectures to further improve performance and scalability in Arabic text analysis tasks.

2) Performance metrics comparison:

- Accuracy: Our approach achieves accuracy ranging from 78.48% to 97.13% across various experiments and classifiers.
- Precision, Recall, F-measure: These metrics consistently show high performance, indicating robustness in negation detection.

3) Comparison with recent works:

- Existing Approaches: Previous studies on negation detection in Arabic texts have predominantly focused on rule-based systems or simple machine learning models with basic feature sets.
- Advancements in Our Approach: Our work stands out due to several factors:
  - Advanced Feature Sets: We explore comprehensive feature sets combining lexical and structural features, which significantly enhance accuracy and F-measure.
  - Machine Learning Models: Leveraging state-of-the-art classifiers such as DT, our approach demonstrates superior performance compared to traditional approaches.
  - Experimental Rigor: We conduct extensive experiments across multiple feature combinations and classifiers, ensuring robust evaluation and comparison.

4) Discussion on novelty and contribution:

- Innovative Feature Engineering: By integrating structural features (e.g., negation words count, length, punctuation) with traditional lexical features, we enhance the model's ability to capture nuanced linguistic cues in Arabic.
- Classifier Performance: The DT classifier, in particular, shows exceptional results, achieving up to 97.13% accuracy and F-measure. This underscores the effectiveness of DT in handling complex patterns in Arabic text.
- Impact of Feature Combinations: Our experiments demonstrate that combining features like negation words count, length, and part-of-speech tags with lexical features substantially improves classification accuracy.
- Generalizability: The robustness and high performance of our approach indicate its potential applicability beyond negation detection to other sentiment analysis tasks in Arabic reviews.

In conclusion, our study makes significant strides in Arabic negation detection by leveraging advanced machine learning techniques and comprehensive feature sets. The results underscore the effectiveness of our approach in achieving high accuracy and robust performance metrics compared to existing methods. Moving forward, further research could explore ensemble methods or neural architectures to continue pushing the boundaries of performance in Arabic text analysis tasks.

D. Challenges in Determining Negation

Detecting negation in sentences poses significant challenges beyond explicit negation markers such as "do not." This subsection explores the complexities involved in understanding negation nuances, considering context, subtle cues, and implicit negations in Arabic reviews.

1) Implicit negation

Implicit negation does not use explicit negation terms but relies on context, structure, or specific linguistic cues to convey negated meaning. Examples include:

- Interrogation, Condition, and Wishing Styles: Sentences like "I hope it doesn't rain today" or "I feel that this supposedly heavy workload is actually quite manageable" use language that suggests a negated interpretation without employing overt negation words [20, 28, 30].

Examples:

- "أتمنى لو تصرف وتعامل موظف الاستقبال كان ودودا والغرف نظافتها وصيانتها جيدة."

(I wish the receptionist was friendlier, and the rooms were clean and well-maintained.)

- "من قال إن الإطلالة على البحر رائعة؟"

(Who said the sea view is great?)

2) Fake negation

Certain negation terms in Arabic can have alternative usages unrelated to negation, such as in condition, interrogation, and wondering styles. This phenomenon complicates sentiment analysis as these terms do not always reverse the sentiment polarity.

Examples:

- "ما أجمل الإقامة في هذا الفندق، فهو يتميز بخدمته الممتازة والغرف المريحة والنظيفة".

(How wonderful it is to stay in this hotel; it stands out with its excellent service and comfortable, clean rooms.)

- "ما به عيب سوى النظافة والخدمة".

(There is nothing wrong with it except the cleanliness and service.)

3) Neutral targets

Negation can also be used to express neutrality, indicating the absence of a specific condition or attribute without inherently conveying positive or negative sentiment.

Example:

- "لم تكن الإقامة سيئة ولكنها لم تكن رائعة أيضًا. كانت تجربة عادية بشكل عام دون متميزات خاصة".

(The stay wasn't bad, but it also wasn't great. It was an overall average experience without any special highlights.)

4) Double negation

In Arabic, double negation is used for emphasis or intensification of negation rather than to convey a positive meaning. It involves the use of two negative particles within a sentence:

Examples:

- "لا أستطيع أن لا أنام".

(I can't not sleep.)

- "الإقامة لم تكن لا تستحق السعر المدفوع. لم لم يكن هناك أي تجربة إيجابية تبرر القيمة المالية".

(The stay wasn't worth the price paid. There wasn't any positive experience to justify the value for money.)

Understanding these nuances is crucial for developing accurate negation detection systems in Arabic reviews. Traditional and deep learning approaches must effectively handle these complexities to enhance their performance in sentiment analysis tasks.

V. Conclusions

In this study, we introduced a comprehensive methodology for detecting negation by classifying "negated positive" and "positive" reviews in user-generated Arabic hotel reviews. Our approach involved several stages, including data collection, preprocessing, feature extraction, supervised learning classification, and evaluation. The main findings of our study are as follows:

- Dataset Composition: We compiled an extensive corpus of 84,000 Arabic opinion reviews from well-known online platforms specializing in Arabic economic content, such as tripadvisor.com, booking.com, and agoda.ae. This dataset was evenly split between 42,000 "negated positive" reviews and 42,000 positive reviews, covering both Modern Standard Arabic (MSA) and Dialectal Arabic (DA), as well as mixed forms.
- Feature Effectiveness: Our approach leveraged both lexical and structural features, such as the number of negation words, length, punctuation-based features, and Part-of-Speech (PoS) tags. These features proved to be effective in the supervised machine-learning process for detecting negation in Arabic hotel reviews.
- Machine Learning Classifiers: We evaluated the performance of three machine learning classifiers—Naïve Bayes (Kernel) (NBK), Decision Tree (DT), and K-Nearest Neighbours (KNN)—in detecting negation in Arabic hotel reviews.
- Performance Metrics: To gauge classifier performance, we used key metrics including accuracy, precision, recall, and F-measure. The outcomes of our experiments demonstrated that all classifiers exhibited highly comparable performance in identifying negation, with only marginal deviations in their performance metrics.
- Top Performer: The Decision Tree (DT) classifier consistently emerged as the top performer among the classifiers tested. It achieved an exceptionally high overall accuracy rate of 97.13%, surpassing established benchmarks in Arabic text processing. This indicates the potential for practical applications of our approach in real-world scenarios.
- Significance: The results of our study hold particular significance within the realm of Arabic text processing. The high accuracy rates achieved, especially by the DT classifier, underscore the feasibility and practicality of our methodology for detecting negation in Arabic hotel reviews. This contributes to the broader field of sentiment analysis and natural language processing for Arabic texts.

In summary, our study has demonstrated the effectiveness of various machine learning classifiers in detecting negation within Arabic hotel reviews, with the DT classifier achieving the best performance. These findings highlight the potential of advanced machine learning techniques in improving the accuracy and reliability of sentiment analysis in Arabic text processing.

Concerning our upcoming research directions, we have the intention of implementing this approach on social networking platforms like Twitter and Facebook. This choice stems from the striking resemblance in the fundamental structure of these platforms. Furthermore, we plan to explore the utilization of lexicons and incorporate new features to enhance our capability to distinguish more effectively between "negated positive" reviews and positive reviews. In our forthcoming investigations, we are also planning to assess the effectiveness of our approach using datasets that exhibit imbalance and real-world data, alongside an evaluation of its performance on more extensive corpora spanning various domains.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

The authors confirm contribution to the paper as follows: A.S.A.: Conceptualization, Methodology, Data curation, Investigation, Software, Visualization, Validation, Writing-Original draft preparation, Writing-Reviewing and Editing; M.A.A.: Supervision, Validation, Writing-Reviewing and Editing. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

[1] L. Burbach, P. Halbach, M. Ziefle, and A. C. Valdez, "Opinion formation on the internet: The influence of personality, network structure, and content on sharing messages online," *Frontiers in Artificial Intelligence*, vol. 3, 2020. doi: 10.3389/frai.2020.00045

[2] A. S. Abuhammad and M. A. Ahmed, "Negation detection techniques in sentiment analysis: A survey," *International Journal of Applied Science and Engineering*, vol. 20, no. 2, pp. 1–7, 2023. doi: 10.6703/ijase.202306_20(2).003

[3] Definition of Negation. (June 2023). [Online]. Available: https://www.lexico.com/definition/negation

[4] Negation Definition and Meaning. (June 2023). [Online]. Available: https://www.collinsdictionary.com/dictionary/english/negation

[5] D. M. Hussein, "A survey on sentiment analysis challenges," *Journal of King Saud University—Engineering Sciences*, vol. 30, no. 4, pp. 330–338, 2018. doi: 10.1016/j.jksues.2016.04.002

[6] S. Mohammad, "A practical guide to sentiment annotation: Challenges and solutions," in *Proc. the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2016. doi: 10.18653/v1/w16-0429

[7] Four Sentiment Analysis Accuracy Challenges in NLP | Toptal®. (June 2023). [Online]. Available: https://www.toptal.com/deep-learning/4-sentiment-analysis-accuracy-traps

[8] U. Farooq, "Negation handling in sentiment analysis at sentence level," *Journal of Computers*, pp. 470–478, 2017. doi: 10.17706/jcp.12.5.470-478

[9] I. G. Councill, R. Mcdonald, and L. Velikovich, "What's great and what's not: Learning to classify the scope of negation for improved sentiment analysis," in *Proc. the Workshop on Negation and Speculation in Natural Language Processing*, Stroudsburg, USA, 2010, pp. 51–59.

[10] P. Mukherjee, Y. Badr, S. Doppalapudi, S. M. Srinivasan, R. S. Sangwan, and R. Sharma, "Effect of negation in sentences on sentiment analysis and polarity detection," *Procedia Computer Science*, vol. 185, pp. 370–379, 2021. doi: 10.1016/j.procs.2021.05.038

[11] A. El-dine and F. El-zahraa, "Sentiment analyzer for Arabic comments system," *International Journal of Advanced Computer Science and Applications*, vol. 4, no. 3, 2013. doi: 10.14569/ijacsa.2013.040317

[12] S. S. Alotaibi, "Sentiment analysis in the Arabic language using machine learning," Ph.D. dissertation, 2015. https://mountainscholar.org/handle/10217/167091

[13] Arabic-Wikipedia. (June 2023). [Online]. Available: https://en.wikipedia.org/wiki/Arabic

[14] World Internet Users' Statistics and 2023 World Population Stats. (June 2023). *Internet World Stats* [Online]. Available: https://www.internetworldstats.com/stats.htm

[15] O. Alharbi, "Negation handling in machine learning-based sentiment classification for colloquial Arabic," *International Journal of Operations Research and Information Systems*, vol. 11, no. 4, pp. 33–45, 2020. doi:10.4018/ijoris.2020100102

[16] A. Funkner, K. Balabaeva, and S. Kovalchuk, "Negation detection for clinical text mining in Russian," *Studies in Health Technology and Informatics*, pp. 43–47, 2020.

[17] S. M. Jiménez-zafra, N. P. Cruz-díaz, M. Taboada, and M. T. Martín-valdivia, "Negation detection for sentiment analysis: A case study in Spanish," *Natural Language Engineering*, vol. 27, no 2, pp. 225–248, Jul. 2020. doi: 10.1017/s1351324920000376

[18] A. Mahany, M. M. Fouad, A. Aloraini, A., H. Khaled, R. Nawaz, N. R. Aljohani, and S. Ghoniemy, "Supervised learning for negation scope detection in Arabic texts," in *Proc. IEEE 10th International Conference on Intelligent Computing and Information Systems (ICICIS)*, Cairo, Egypt, 2021, pp. 177–182.

[19] H. K. Aldayel and A. M. Azmi, "Arabic tweets sentiment analysis —A hybrid scheme," *Journal of Information Science*, vol. 42, no. 6, pp. 782–797, Jul. 2016. doi: 10.1177/0165551515610513

[20] RapidMiner | Amplify the impact of your people, expertise and data. (June 2023). [Online]. Available: http://rapidminer.com/

[21] Lexical analysis. Wikipedia. (June 2023). [Online]. Available: https://en.wikipedia.org/wiki/Lexical_analysis#Tokenization

[22] R. Feldman and J. Sanger, *The Text Mining Handbook*, Cambridge University Press, 2007.

[23] A. Pak, and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," in *Proc. the 7th International Conference on Language Resources and Evaluation (LREC 2010)*, Valletta, Malta, 2010, vol. 10, pp. 1320–1326.

[24] C. Silva and B. Ribeiro, "The importance of stop word removal on recall values in text categorization," in *Proc. The International Joint Conference on Neural Networks*, Portland, OR, USA, 2003, vol. 3, pp. 1661–1666.

[25] N. A. Abdulla, N. A. Ahmed, M. A. Shehab, M. Al-ayyoub, M. N. Al-kabi, and S. Al-rifai, "Towards improving the lexicon-based approach for Arabic sentiment analysis," *International Journal of Information Technology and Web Engineering*, vol. 9, no. 3, pp. 55–71, Jul. 2014. doi: 10.4018/ijitwe.2014070104

[26] L. Larkey, L. Ballesteros, and M. Connell, "Improving stemming for Arabic information retrieval: Light stemming and co-occurrence analysis," in *Proc. the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrievalnal*, Tampere, Finland, 2002, pp. 275–282.

[27] M. Saad, "The impact of text preprocessing and term weighting on Arabic text classification," M.S. dissertation, Department of Computer Engineering, The Islamic University-Gaza, 2010.

[28] N. Farra, E. Challita, A. Assi, and H. Hajj, "Sentence-level and document-level sentiment mining for Arabic texts," in *Proc. IEEE International Conference on Data Mining Workshops (ICDMW)*, 2010, pp. 1114–1119.

[29] J. Reitan, J. Faret, B. Gambäck, and L. Bungum, "Negation scope detection for twitter sentiment analysis," in *Proc. the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2015, pp. 99–108.

[30] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. Wiley Interscience, 2001.

[31] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, Elsevier, 2011.

[32] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*, Elsevier, 2011.