


Negation Handling in Machine Learning-Based Sentiment Classification for Colloquial Arabic

Omar Alharbi, Jazan University, Jizan, Saudi Arabia

 <https://orcid.org/0000-0003-4149-6068>

ABSTRACT

One crucial aspect of sentiment analysis is negation handling, where the occurrence of negation can flip the sentiment of a review and negatively affects the machine learning-based sentiment classification. The role of negation in Arabic sentiment analysis has been explored only to a limited extent, especially for colloquial Arabic. In this paper, the authors address the negation problem in colloquial Arabic sentiment classification using the machine learning approach. To this end, they propose a simple rule-based algorithm for handling the problem that affects the performance of a machine learning classifier. The rules were crafted based on observing many cases of negation, simple linguistic knowledge, and sentiment lexicon. They also examine the impact of the proposed algorithm on the performance of different machine learning algorithms. Furthermore, they compare the performance of the classifiers when their algorithm is used against three baselines. The experimental results show that there is a positive impact on the classifiers when the proposed algorithm is used compared to the baselines.

KEYWORDS

Arabic Sentiment Analysis, Colloquial Arabic Language, Machine Learning, Negation, Sentiment Lexicon

INTRODUCTION

Sentiment analysis or opinion mining is the process of automatically identifying the opinions expressed at the level of a word, sentence, or document. In recent years, Sentiment Analysis has received much attention from researchers, and considerable progress has been achieved for different languages, especially for English. However, as this work concerns with the Arabic language, the task of sentiment analysis is still limited. The challenges that face Arabic sentiment analysis are related to the inflectional nature of the language itself. El-Beltagy and Ali (2013) highlighted many issues of sentiment analysis in the Arabic language such as the presence of dialects, lack of Arabic dialects resources and tools, limitation of Arabic sentiment lexicons, using compound phrases and idioms, etc. In general, the Modern Standard Arabic (MSA) and colloquial Arabic are the most commonly used forms of Arabic language in social networks, blog, and forums.

DOI: 10.4018/IJORIS.2020100102

This article, originally published under IGI Global's copyright on October 1, 2020 will proceed with publication as an Open Access article starting on February 2, 2021 in the gold Open Access journal, International Journal of Operations Research and Information Systems (converted to gold Open Access January 1, 2021), and will be distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

One of the several challenges that face sentiment analysis, in general, is the negation. In sentiment analysis, negation words can reverse the meaning of a sentence; as a result of which the sentiment orientation would be changed. For example, negation words (such as “not”) flip the sentimental orientation of the terms (such as “good”). According to Wiegand, Balahur, Roth, Klakow, and Montoyo (2010), negation is a complicated issue, and it is highly relevant for sentiment analysis. This is a complicated issue because it requires detecting the negation words and then identifying the affected words (which are called negation scope) based on either syntactic or semantic representations (Ballesteros et al., 2012). For sentiment classification, there are two major approaches used in the literature. The semantic orientation approach in which sentiment lexicons and other linguistics resources are used to compute the sentiment polarity of a given sentence based on the polarity of its words (Taboada, Brooke, Tofiloski, Voll, & Stede, 2011). The other approach is a machine learning-based sentiment classification, which uses annotated data from which a set of features is extracted as a training data used by a classifier to build a model for predicting the classes of a testing data using one of the machine learning algorithms (Pang, Lee, & Vaithyanathan, 2002). In the literature of sentiment analysis, machine learning approach has outperformed the semantic orientation approach in several aspects (Morsy, 2011). However, the performance of machine learning algorithms would be affected by the presence of negation terms (Jia, Yu, & Meng, 2009; Wiegand, Balahur, Roth, Klakow, & Montoyo, 2010; Zhang, Ferrari, & Enjalbert, 2012). In fact, determining the sentiment polarity requires taking linguistic context (such as negation) into account instead only simple presentations such as Bag-of-Words (BOW) and n-grams. Many studies have been published to address the problem of detecting negation words and the negation scope to improve the performance of machine learning algorithm-based sentiment classification such as in (Jia, Yu, & Meng, 2009; Morante & Daelemans, 2009; Pang & Lee, 2004; Polanyi & Zaenen, 2006).

Likewise, negation plays a crucial role in performance of sentiment analysis for the Arabic language, whether in MSA or dialects (Duwairi & Alshboul, 2015; Morsy, 2011). However, the problem of negation algorithms has been less explored in Arabic sentiment analysis using machine learning. Most of the previous work on Arabic sentiment classification used various machine learning algorithms without considering the effect of negation on the classification performance. Not taking negation into consideration would create a similar representation of two sentences like “مفللا اده بجا انا” (I like this movie) and “مفللا اده بجا ال انا” (I do not like this movie), although the first one hold a positive sentiment polarity while the second one holds a negative sentiment polarity. That would negatively affect the performance of the classifiers used for sentiment analysis.

In Arabic sentiment analysis, colloquial texts were less addressed compared to MSA. This is presumably due to the availability of different linguistic resources for the MSA language compared to colloquial Arabic language. However, when it comes to social networks or reviewing products, most people use their dialects instead of MSA. Therefore, in this research, we are concerned with handling the negation problem in the colloquial Arabic texts. Unlike MSA, the colloquial Arabic is not restricted with grammatical rules through which the negation can be simply detected. The poor grammatical structure and the lack of resources tools such as morphological parser, part of speech (POS) taggers, and stemmers for colloquial Arabic made the task of negation handling is challenging.

Previous studies on Arabic sentiment analysis- as will be explained in section 2- adhere to more or less similar approaches to handle the negation problem. These approaches dealt with the problem in different ways; a statistical way in which the frequency of the negation terms in a given sentence is represented as a feature, or another way by negating a window of words whenever preceded by a negation word, by marking them with a negation tag as suggested by Das and Chen (2001). By using these approaches, negation cannot be properly modeled, that can be explained by the following example, a sentence like “نوكي ام بقرأ نم لماعتلا” (dealing with customers one of the finest) would result with 1 occurrence of “ام” based on the former approach, despite the term “ام” in this example is a relative pronoun not a negation word. Based on the latter approach, in a

sentence like “ادج قنداه سكلاب جاعزا يف ام” (there is no noise, on the contrary, it was very quiet), the words after “ام” would be marked with a negation, although the only word which supposed to be affected is (جاعزا) “noise”. This would result with new features created that negatively affect the performance of the sentiment classification. Nevertheless, we present these models as baselines to be compared with the proposed algorithm. On the contrary, the proposed algorithm aims to detect only the affected words as some opinionated words might not be affected even though they are within the scope of a negation term. Another approach used to capture negation is using higher-order n-grams, such as using bi-gram in the work of Pang, Lee, and Vaithyanathan (2002). Although this approach is convenient, this would fail in cases in which the affected words are at a distance from the negation words. For instance, in a sentence like “يكاڤ عيش يا معطما اذهب دجوي ال” (there is no anything in this restaurant is delicious), the algorithm needs 6-gram to capture negation (... يكاڤ ال) “no...delicious”, and using such high order n-grams would lead to very sparse representation that makes the learning from training data is harder.

As a main contribution, we propose an algorithm that can detect and handle the negation problem in the colloquial Arabic reviews to improve the performance of the machine learning-based sentiment classification. The author also examines the effect of the proposed algorithm on four of the most common classifiers used in sentiment analysis; they are Support Vector Machine (SVM), Naïve Bayes (NB), k-nearest neighbor (KNN), and Logistic Regression. Additionally, a comparison is carried out between the classifiers when our algorithm is used and three baseline models that differ in their methods of determining the negation scope. The proposed algorithm uses crafted rules, linguistic knowledge, and sentiment lexicon. The rules were crafted based on observing many cases of negation in colloquial reviews. It detects the negation words, like (ال، وم) “no, not, not”, and then mark the opinionated words that might be affected within a predefined window length of words. These rules do not rely on grammatical knowledge about the relationships between different constituents, as there are no standard grammatical rules to dialectal texts. A major challenge in this respect is determining the sequence of words in the sentence that might be affected (negation scope) by a negation term. Unlike the Arabic language, several approaches based on various aspects of contents have been presented to address this issue in the English language. These approaches require an annotated negation dataset which is not available for colloquial Arabic language. This issue is beyond the scope of this paper; therefore, we solely use a predefined window length of five words that directly follow a negation word.

The remainder of the paper is organized as follows. Section 2 presents related work that considered negation in Arabic sentiment analysis. In section 3, we introduce the methodology through which the sentiment classification including negation handling is performed. We discuss experimentations and results in section 4. Finally, Section 5 presents the conclusion of this work.

RELATED WORK

Many works have explored the negation problem in detail in the English and other languages (Amalia, Bijaksana, & Darmantoro, 2018; Gautam, Maharjan, Banjade, Tamang, & Rus, 2018; Mittal, Agarwal, Chouhan, Bania, & Pareek, 2013; Villalba-Osornio, Pérez-Celis, Villasenor-Pineda, & Montesy-Gómez; Zou, Zhu, & Zhou, 2015), whereas few studies have addressed this issue in the Arabic language as this field is still at an early stage. In this section, we explore how the negation problem has been addressed in previous studies of Arabic sentiment analysis. Several studies in sentiment classification have employed different machine learning algorithms. Unfortunately, a few only have considered the negation problem when the terms are turned into features for the learning process. One of the common methods to include the negation into the features representation is to identify all the words whenever they are preceded by a negation word within either a fixed window size, until the first punctuation, or until the end of a sentence, and then have them marked a negation tag, such as in the studies of (Abdulla, Ahmed, Shehab, & Al-Ayyoub, 2013; Adouane & Johansson, 2016; Al-

Obaidi & Samawi, 2016; Duwairi, Marji, Sha'ban, & Rushaidat, 2014; El-Halees, 2011). However, none of these studies reports how the sentiment classification was affected by negation or sufficient details about how they handled the problem. According to Wiegand, Balahur, Roth, Klakow, and Montoyo (2010), this method cannot properly model the negation scope, as that would lead to tag words which are not supposed to be. Consequently, one word would be treated as two different words, then feature space increases.

Another method to deal with negation in the Arabic sentiment classification is the frequency or presence of negation words as a feature of the phrase in a supervised classifier such as in the studies of (Adouane & Johansson, 2016; Al-Harbi, 2017; Farra, Challita, Assi, & Hajj, 2010; Hamouda & El-taher, 2013). In this method, the count of sentiment words also can be used as features after reducing the negated words form a polarity type and consider it with the opposite polarity type. One way to obtain the knowledge about sentiment words is by using a sentiment lexicon which contains a list of opinionated words attached with polarity labels. However, this method does not consider how and what the words in the negation scope that should be affected by the negation terms and that would lead to lower performance.

Other few studies have investigated negation in Arabic sentiment analysis. For example, in the work of Elhawary and Elfeky (2010), they propose a technique that handles negation embedded in Arabic reviews. The authors do not mention whether the reviews written with MSA or dialect. They assume that the negation scope is all words whenever preceded with negation terms until the end of the sentence. As explained before, such a method will lead to issues like negating words that should not be affected by the negation terms. Furthermore, no details provided about the negation words they used except its number, which was 20 words, nor how they dealt with the cases in which the negation words do not have the negation sense. As well as, the effect of negation on sentiment classification is not reported. Another work also introduced by Mostafa (2017), which focus on handling negation in both MSA and dialectal Arabic. The author did not mention details about the dataset and dialect used in this work. He provided details about an algorithm that deals with the negation problem; however, what proposed is not different from the traditional methods that inverse any polarity expression that follow a negation term. The author also considered only one case in which exceptional negation is used. The reported results show an improvement after applying the exceptional negation algorithm to the classifiers SVM, NB, and K-NN. On the contrast, our algorithm uses sentiment lexicon to detect only the affected polarity terms within a fixed window size, and many cases appeared in the texts were investigated to be handled.

The work of Duwairi and Alshboul (2015) also focused on the negation problem. In this work, they introduce an unsupervised sentiment analysis for MSA language that includes a morphological framework for negation. They propose a treatment of negation by using a set of rules derived from formal linguistic knowledge. The negation words are categorized into two groups, the first one include (نل، مل، ال، ام) which affect only the verb that appears immediately after them, and the second group contains (سئل) which affect only the two nouns following it. ArSenl lexicon and an Arabic morphological analyzer were used to assign the sentimental value and POS for the terms, respectively. Unfortunately, these rules cannot be applied to the texts in our work due to the presence of dialect that does not abide by the same linguistic rules. Furthermore, the authors did not provide any details about the experiment or the evaluation results for their approach.

METHODOLOGY

The aim of this research is to handle the negation problem to improve the sentiment classification for the colloquial Arabic reviews. This section describes the methodology through which the proposed algorithm was developed. The following sections introduce the proposed algorithm, necessary resources, and tools.

Dataset

To train the classifier, we need an annotated dataset. For the purpose of this work, the author used a publicly available dataset for Jordanian dialect presented by Al-Harbi (2017). The dataset is annotated on the document level, and it considers only two polarity classes, which are positive and negative. To balance the dataset, we randomly selected 2400 reviews, of which 1200 were positive, and 1200 were negative. The data consists of MSA and colloquial Jordanian reviews about various domains (restaurants, shopping, fashion, education, entertainment, hotels, motors, and tourism).

Pre-Processing

The pre-processing stage included removing noise from data, normalization, and tokenization. The process of removing noise from data includes removing misspellings, repeated letters, diacritics, punctuations, numerals, English words, and elongation. After that, a normalization process was applied to particular letters, for example the letters (أ, إ, آ) were converted to (ا), the letters (ي, ع) were converted to (ي), the letter (ة) was converted to (ه), and finally the letter (ة) was converted to (و). Tokenization is the process of dividing a given text into a set of words (tokens) which are separated by spaces.

Sentiment Lexicon

In this work, the author adopted the dialectical lexicon developed by Al-Harbi (2017). This lexicon consists of 3400 sentiment-bearing words labeled with either positive or negative polarities. The terms were extracted from a text written in both MSA and colloquial language. This lexicon is used to trace the words affected by negation words within the negation scope. During the analysis process, the algorithm will detect all the polarity words and decide whether the negation is meant to positive or negative words, and based on that, only affected words will be reversed.

Negation Terms List

The author manually collected the most common negation terms used in the reviews and stored them in a list, including different morphological forms of some words. The negation list contains 50 terms, including the terms used in both types of Arabic, MSA such as (سبيل، مل) and the dialectal words. In Jordanian dialect, negation is expressed with different terms from MSA. For example, the terms (وهم، شيفم، شف، شوهم، شم، وم) were used in the collected texts. Another way used to negate words is using terms like (تميال، ويفام، اهيفام، اهيبام، يفام) due to that the people tend to not space between the negation terms and the following word. We treated such cases as one expression that belongs to the negation words. In this work, if a negation term is detected in the review, the following words within a window length of 5 words will be checked against the sentiment lexicon to decide if they need to be reversed by marking them as negated words. On the other hand, there are several cases in which the negation terms were detected, but not followed by sentimental words, for instance, the review (قبيلز نملا تاودألا مسق يفام) “there is no home appliances section”, in these cases, the algorithm will not mark any word within the scope with a negation tag.

Negation Handling

The main objective of the paper is to address negation in colloquial Arabic reviews to improve sentiment classification. This section describes the proposed algorithm to handle this problem. The algorithm was developed using Python 3.0 programming language, see Figure 1. The input to our algorithm is a review with one or more occurrences of negation terms and output the review with negated polarity words if detected within the negation scope. First of all, we introduce the mechanism of detecting the negation terms and negation scope, which is simply tracing the negation terms within a given review based on the predefined negation terms. Then, if sentimental words are detected within the negation scope, the words will be marked with a negation tag, for instance,

(معطماً اذہ >حباً! <ال) “I don’t like_! this restaurant”. Each negation term is assumed to have a scope of negation effect. In this work, the negation scope is the five words that directly follow the negation term. Determining the negation terms is not an easy task, particularly in the Arabic language since sometimes a negation term in a review does not have the negation sense, or might affect one sentimental polarity without the other. Knowing that, there is no morpho-syntactic tools can be used to the colloquial Arabic, made detecting such exceptions even complicated task. To this end, many cases have been analyzed to come up with rules that can detect such exceptions. In this section, we summarize several cases of how negation terms used in the colloquial Arabic reviews, from which we crafted the required rules to detect negation properly.

Case1: A sentence has a negation word followed by an exceptional word (إلا) “but, or except” and polarity expression within the negation scope, and the index of the exceptional word is greater than the index of negation word and less than the index of polarity term like in the sentence (فارتحالو هسيوكلا هلماعملا الا تيقل ام هحارصب) “Frankly, we did not find anything but proper treatment and professionalism”. In this case, the negation word is used to emphasize whatever the polarity comes after the exceptional word which is positive polarity in this sentence expressed by (فارتحالا, هسيوكلا) “proper, professionalism”. Therefore, the algorithm will not mark the polarity words as negated.

Case2: Another phenomenon used commonly in the texts is the use of superlative and comparative words preceded with negation words to express the sentiment as in the sentence (ديج لايفتساو فيظن انهم نلحا يفام هيرجيت) “There is no more beautiful than this experience; it was a clean and good reception”. The negation word (يفام) “there is no” followed by the word (نلحا) “more beautiful” were used to express positive sentiment, so expectedly any sentimental term comes after those agree with the same polarity and that obvious with the words (ديج، فيظن) “clean, good” that also express positive sentiment. Another example with a negative sentiment (نييادك) “there is no worse than such people, liars”, where is the polarity of the word (نسا كيه نم أوسا يفام) “liars” agrees with the polarity of the word (أوسا) “worse” and the negation here would not be appropriate. As can be noted in this case, the index of superlative and comparative words is always greater than the index of negation word and less than the index of polarity term. In this case, the algorithm will discard negating the polarity word, and in order to do that, given that we decided to not use any morphological analyzer, we collected and stored the most common used comparative and superlative words such as (نعلاً، أوساً، بقراً، مخفأً، نسحاً، لمجأً)، (لضفأً، نلحاً، عوراً).

Case3: A sentence has two or more sentimental words with different polarities (positive and negative), which fall into the negation scope like in the sentence (قرملا ب خسو ناكملا ولح شم) “Not a lovely place, it is very filthy”. The presence of a negation term in a sentence does not mean that all its polarity words should be affected. As we can see in the example, there are two sentimental words within the negation scope (ولح) “lovely” which expresses positive sentiment and (خسو) “filthy” which expresses negative sentiment. In this case, the algorithm will detect the polarity of the first sentimental word occurs after the negation term which is in the above sentence (ولح), then will negate only the words that fall into the same polarity within the scope and discarding any other polarity.

Case4: A sentence has the negation term (ام) that holds different senses other than the negation, such as interrogative or relative pronoun. For instance, (ناكملا ريغنو دكتتن مهيلع حورن ام لك) “Every time we visit them, we got miserable, and we then change the place”, based on the discourse context, the word (ام) is a relative pronoun that does not has a negation effect on the negative sentiment of the word (دكتتن) “got miserable”; however, the capability to recognize such cases is hard without a morpho-syntactic analyzer. As mentioned before, we cannot use such analyzer since the available ones have been trained only on MSA. Therefore, we collected and stored all the words that used frequently before or after (ام) when it does not express the nega-

tion sense. Table 1 shows most the cases of (ام) as not a negation term, whenever, these cases detected the algorithm will ignore negating any polarity term within the scope.

Table 1. Words might appear after or before the negation word “ام”

Negation Term	Cases
Before “ام”	ام لبق، ام دعب، ام اعون، ام يز، ام نل، ام نودب، ام يا، ام وش، ام لوأ، ام دحل، ام ذنم، ام لدب، ام دق، ام بسح، ام لثم، ام لثم، ام لك
After “ام”	هلا عاش ام، لبق ام، دعب ام

Case5: A sentence has the negation term (ريغ) which in some cases does not have the negation effect on the words like in the sentence (تجعز ملا نكامألا نع ريغ تلالناعل قيسانم نكامأ) “These places are suitable for families; they are different from the noisy places”. The word (ريغ) in the sentence means “different from”, and it cannot play the role of the negation on the polarity word (تجعز ملا) “noisy”. In this case it is hard to recognize the word without morphological knowledge, however, the proposed algorithm can handle this case based on knowledge of the words used frequently whether before or after (ريغ). Those words were observed and collected from the dataset to be fed to the algorithm, Table 2 shows the words.

Table 2. Words might appear after or before the negation word “غير”

Negation Term	Cases
Before “ريغ”	ريغ ال
After “ريغ”	لكش ريغ، كيه ريغ، متنا ريغ، كلذ ريغ، هنا ريغ، ونا ريغ، نع ريغ

Case6: Other cases were observed in which the negation terms do not have the negation sense. To enable the algorithm to detect such cases, we collected the words that might frequently appear before or after the negation terms in these cases as knowledge to guide the algorithm to decide whether it is a negation word or not. Table 3 shows the cases we collected along with examples.

Features Representation

In any machine learning approach-based classification, a suitable text representation model is required. This model is often called a vector model or feature model, which is represented by a matrix of term weights. The work of Al-Harbi (2019) has already examined the best text representation for this dataset. Furthermore, the effect of stop words removal, weighting schemes, and stemming (light stemming and root stemming) on the performance of the classifiers were evaluated. The result was a combination of uni-grams, Term Frequency-Inverse Document Frequency (TF-IDF), and stop words removal gives the best performance.

Sentiment Classification

The goal of the classification is to categorize input data into predefined classes and produce a model based on training data, which predicts the target values of the test data. This work concerns only with two classes; they are positive and negative. In this work, four machine learning algorithms which

Table 3. Words might appear after or before the negation words “م، ل، لم، مش، هو، ل، لم”

Negation Term	Before	After	Example
شم Not	إذا If	-	مهنسحا شم اذا ندرالاب. سارعالا نيمظنم لضافا Good wedding organizers in Jordan, if not the best.
وم Not	-	لثم، لثم like	نبيلاغ تالحملا يقاب لثم وم. هيعيبط ادج راعسا Prices are reasonable, not like other expensive stores.
ال Not	-	دب Must	جاعزإلا هنم دب ال يعني. ينالعا لصاصف امياد Always ads during the movie, is it must be annoying?
مل Not	نا If	-	ندرألاب لضافلا نكت مل. نا نامعب تابتكملا نسأ The best library in Amman, if not the best in Jordan.

represent diverse approaches were used to explore the effect of negation handling on colloquial Arabic sentiment classification, namely, SVM, NB, KNN, and Logistic Regression.

EXPERIMENT AND EVALUATION

This section describes the experiments undertaken to evaluate the performance of the chosen machine learning algorithms on colloquial Arabic sentiment classification when the proposed algorithm is used.

Experiment Settings

Different experiments were performed to examine the effect negation on the colloquial Arabic sentiment analysis. To accomplish these experiments, we used Rapidminer, which is a software platform that includes a valuable set of machine learning algorithms and tools for data and text mining. As mentioned earlier, the dataset includes reviews which were annotated on the document level, and consist of 2400 reviews of which 1200 were positive, and 1200 were negative. Based on an investigation into the dataset to compute the percentage of the negation, it was found that 47% of the reviews contain explicit negation terms of which 74% were negative, and 26% were positive. It is clear that users tend to use negation terms more when they express negative opinions. The experiments were implemented using four classifiers, namely, SVM, NB, Logistic Regression, and K-NN. Regarding the SVM classifier, we used LIBSVM (Chang & Lin, 2001) with a kernel type of linear as it empirically gave the best performance in our previous works (Al-Harbi (2017); Al-Harbi, 2019). Also, the author investigated K-NN algorithm to find the value of K with which it gives the best performance, based on that the value of K was set to 50. Another issue arises when it comes to using machine learning classifiers is tuning the hyperparameters which can lead to different results

Figure 1. Algorithm for negation handling

Algorithm 1: Negation Handling.

```
Input: A Review
Output: A Review With Negated Polarity Terms If
        Detected
1  Read Review
2  for word in range(len(the review)):
3      if the word is a negation term, and not
4  Case1,
5      and not Case2, and not Case3, and not Case4
6      and not Case5, and not Case6:
7          for i in range(0, 5):
8              if the next word in the sentiment
9              lexicon:
10                 word = word+"_"
11             else:
12                 Discard and check the next word
13         else:
14             Discard and check the next review
15     close()
```

for the same classifier. However, finding the optimal hyperparameters is not within the scope of this work; therefore, we follow the default settings provided by RapidMiner software. As previously mentioned, TF-IDF weighting scheme is used to represent the uni-grams after removing the stop words from the reviews.

To examine the effect of the proposed algorithm, there is a need to provide a proper baseline to be compared with. The author used the traditional models that have been employed in different related studies as baseline models for this work. In particular, there are three baseline models to be compared with the proposed algorithm. The first one is baseline1 in which the simple uni-gram model is used without considering the negation problem. Secondly, baseline2 in which a uni-gram model is used, considering the negation problem with a negation scope of five words that directly follow a negation term, where, each term within the scope will be tagged with the negation mark. The last one is baseline3, in which a uni-gram model is used with a negation scope includes all the words that follow a negation term until the end of the sentence, where, each term within the scope will be tagged with the negation mark.

Evaluation Metrics

In order to evaluate the performance, the N-fold cross validation was employed with N=10, since it has been widely used in this field as it is a reliable technique for assessment. By using this assessment method, the whole dataset was divided randomly into 10 sets with equal sized samples, where the classifier was trained on 9 sets and the remaining set was used for testing. To measure the performance of the machine learning classifiers, the following evaluation metrics were chosen: Accuracy, Precision, and Recall; see Equations 1, 2 and 3. The accuracy represents the correctness percentage of the model by averaging the correct classifications on the total number of classifications. The precision calculates the accuracy of the classifier in regards to the specific predicted class. The recall is sometimes shows the percentage of the correct predicted classes among the actual class in the data:

$$Accuracy = \frac{TP + TN}{TP + FP + Tn + FN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

where TP indicates a true positive which means the number of the inputs in data test that have been classified as positive when they are really belong to the positive class. TN indicates a true negative which means the number of the inputs in data test that have been classified as negative when they are really belong to the negative class. FP indicate a false positive which means the number of the inputs in data test that have been classified as positive when they are really belong to the negative class. FN indicates a false negative which means the number of the inputs in data test that have been classified as negative when they are really belong to the positive class.

Results

In this section, the author reports the assessment results of the machine learning classifiers that were used to examine the effect of the proposed algorithm. As mentioned above, three baseline models were used for the comparison with the proposed algorithm. Table 4 displays the performance results of the four classifiers when the baseline models the proposed algorithm are used. It can be noticed that all the classifiers give the lowest accuracy, precision and recall when baseline1 is used in comparison with baseline2. That might be fair because in this model, no linguistic knowledge was involved in the learning process. Another notice can be seen; when we compare the recall and precision of the baseline classifiers, it is found that all classifiers give a lower recall percentage. That can be explained by the fact that mentioned earlier, which is the presence of negation in negative reviews more than its presence in positive reviews. In other words, the false negative FN is bigger than the false positive FP, where FN affects the recall metric and FP affects the precision metric. However, the NB classifier was an exception, where it gave a better recall when the baselin1 is used.

In terms of baseline2, SVM, NB, and logistic regression gave better results of accuracy, precision, and recall compared to baseline1. This improvement obtained when negation is considered by marking all the words within the window size of 5 words. In this case, as we can see, there is a considerable improvement in recall which suggests that considering negation positively handled the false negative FN, and that is again because most of negation appears in negative reviews. Conversely, the K-NN classifier is negatively affected by considering negation in baseline2. This can be seen from the dropping of accuracy and recall compared to their values when baseline1 is used. Additionally, despite the improvement obtained by applying baseline2, that would compromise the learning process by adding useless sparse feature space. For instance, 14304 features have been created when baseline2 is used, on the other hand, there were 12074 features used in the learning process when baseline1 is used.

When baseline3 is used, the SVM classifier does not appear to have a significant improvement in terms of accuracy and precision compared to baseline1 and baseline2. However, SVM obtained the best recall, which suggests that considering negation would improve the performance but unfortunately with compromising other aspects. Likewise, considering negation has a negative impact on K-NN, where the performance dropped even less than baseline1. Nevertheless, NB, and logistic regression still give better performance than baseline1, but compared to baseline2, it gave lower results. The low performance of baselin3 can be explained by the issue of sparse representation and its effect on the learning process, where the created features were 18692.

On the other hand, the performance of classifiers using the proposed algorithm showed superiority compared to the baseline models. However, there was an exception; the NB classifier gave a lower performance by a slight percentage compared to baseline2, even though it outperformed baseline1 and baseline3. Another notice worth mentioning that is although the proposed algorithm improved the performance of the SVM classifier, it yielded a little improvement in terms of the accuracy compared to the baseline2. Nevertheless, the algorithm appears to have a significant positive effect on both recall and precision in comparison with all the baselines without compromising each other. Apparently, the

same scenario of SVM happened to the logistic regression classifier, where there was a significant improvement of the performance in comparison with baseline1 and baseline3, on the other hand, there was a slight improvement compared to baseline2. Also, we can notice that both recall and precision using the algorithm yielded the best results compared to the baselines. The proposed algorithm also succeeded to improve the performance of the K-NN compared to the baselines. Although the positive impact on the recall of K-NN, the recall was lower than the precision in all cases. It appears that the results when baseline2 is applied were close to our algorithm, with superiority to our algorithm in terms of recall and precision in most cases. Additionally, in contrast to baseline2 and baseline3, our algorithm avoided creating a sparse representation, which would negatively affect the learning process.

Table 4. Results of proposed algorithm and baseline models

Classifier	Baseline Model	Accuracy	Precision	Recall
SVM	Baseline 1	87.83%	88.35%	87.17%
	Baseline 2	89.08%	88.47%	90.00%
	Baseline 3	87.42%	85.12%	90.75%
	Proposed	89.17%	89.10%	89.33%
NB	Baseline 1	77.83%	76.77%	79.92%
	Baseline 2	80.62%	78.62%	84.25%
	Baseline 3	79.00%	79.01%	79.25%
	Proposed	80.04%	78.44%	83.00%
Logistic Regression	Baseline 1	83.33%	84.17%	82.25%
	Baseline 2	85.29%	87.00%	83.08%
	Baseline 3	83.67%	85.32%	81.42%
	Proposed	85.75%	87.02%	84.17%
K-NN	Baseline 1	86.50%	87.91%	84.67%
	Baseline 2	85.88%	90.29%	80.42%
	Baseline 3	82.75%	87.26%	76.75%
	Proposed	87.75%	89.97%	85.00%

CONCLUSION AND FUTURE WORK

Based on the experimental results, we conclude that using the proposed algorithm for negation handling have a positive impact on machine learning-based colloquial Arabic sentiment classification, yet is far from perfect. The experiments were conducted using SVM, NB, K-NN, and logistic regression, which showed a significant improvement in their performance after applying the negation handling algorithm. The proposed algorithm is rule-based, and the rules were crafted based on observing many cases of negation and simple linguistic knowledge. These rules showed the capability of deciding when the negation should be applied even though the absence of morphological knowledge for colloquial Arabic texts.

In future work, we plan to enable the algorithm to deal with implicit negation that also can negatively affect polarity classification. Another problem that needs to be addressed is that the usage of intensifiers and diminishers, which can change the polarity of words or phrases.

REFERENCES

- Abdulla, N. A., Ahmed, N. A., Shehab, M. A., & Al-Ayyoub, M. (2013). *Arabic sentiment analysis: Lexicon-based and corpus-based*. Paper presented at the conference on applied electrical engineering and computing technologies (AEECT), Amman.
- Adouane, W., & Johansson, R. (2016). *Gulf arabic linguistic resource building for sentiment analysis*. Paper presented at the the Tenth International Conference on Language Resources and Evaluation, Portorož.
- Al-Harbi, O. (2017). Using objective words in the reviews to improve the colloquial arabic sentiment analysis. *International Journal on Natural Language Computing*, 6(3), 1–14. doi:10.5121/ijnlc.2017.6301
- Al-Harbi, O. (2019). A comparative study of feature selection methods for dialectal arabic sentiment classification using support vector machine. *International Journal of Computer Science and Network Security*, 19(1), 167–176.
- Al-Obaidi, A. Y., & Samawi, V. W. (2016). *Opinion mining: Analysis of comments written in arabic colloquial*. Paper presented at the the World Congress on Engineering and Computer Science, San Francisco.
- Amalia, R., Bijaksana, M. A., & Darmantoro, D. (2018). *Negation handling in sentiment classification using rule-based adapted from indonesian language syntactic for indonesian text in twitter*. Paper presented at the Journal of Physics: Conference Series. doi:10.1088/1742-6596/971/1/012039
- Ballesteros, M., Díaz, A., Francisco, V., Gervás, P., De Albornoz, J. C., & Plaza, L. (2012). Ucm-2: A rule-based approach to infer the scope of negation via dependency parsing. *Proceedings of the Sixth International Workshop on Semantic Evaluation*.
- Chang, C.-C., & Lin, C.-J. (2001). Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3), 1–27. doi:10.1145/1961189.1961199
- Das, S., & Chen, M. (2001). *Yahoo! For amazon: Extracting market sentiment from stock message boards*. Paper presented at the the Asia Pacific finance association annual conference, Bangkok.
- Duwairi, R. M., & Alshboul, M. A. (2015). *Negation-aware framework for sentiment analysis in arabic reviews*. Paper presented at the 3rd International Conference on Future Internet of Things and Cloud, Rome. doi:10.1109/FiCloud.2015.115
- Duwairi, R. M., Marji, R., Sha'ban, N., & Rushaidat, S. (2014). *Sentiment analysis in arabic tweets*. Paper presented at the 5th International Conference on Information and Communication Systems, Malacca.
- El-Beltagy, S. R., & Ali, A. (2013). *Open issues in the sentiment analysis of arabic social media: A case study*. Paper presented at the 9th International Conference on Innovations in Information Technology, Abu Dhabi. doi:10.1109/Innovations.2013.6544421
- El-Halees, A. M. (2011). *Arabic opinion mining using combined classification approach*. Paper presented at the International Arab Conference on Information Technology, Zarqa.
- Elhawary, M., & Elfeky, M. (2010). *Mining arabic business reviews*. Paper presented at the the international conference on data mining workshops, Sydney.
- Farra, N., Challita, E., Assi, R. A., & Hajj, H. (2010). *Sentence-level and document-level sentiment mining for arabic texts*. Paper presented at the the international conference on data mining workshops, Sydney. doi:10.1109/ICDMW.2010.95
- Gautam, D., Maharjan, N., Banjade, R., Tamang, L. J., & Rus, V. (2018). *Long short term memory based models for negation handling in tutorial dialogues*. Paper presented at the the Thirty-First International Flairs Conference, Melbourne.
- Hamouda, A. E.-D. A., & El-taher, F. (2013). Sentiment analyzer for arabic comments system. *International Journal of Advanced Computer Science and Applications*, 4(3), 99–103.
- Jia, L., Yu, C., & Meng, W. (2009). *The effect of negation on sentiment analysis and retrieval effectiveness*. Paper presented at the 18th ACM conference on Information and knowledge management, Hong Kong. doi:10.1145/1645953.1646241

- Mittal, N., Agarwal, B., Chouhan, G., Bania, N., & Pareek, P. (2013). *Sentiment analysis of hindi reviews based on negation and discourse relation*. Paper presented at the the 11th Workshop on Asian Language Resources, Nagoya.
- Morante, R., & Daelemans, W. (2009). *A metalearning approach to processing the scope of negation*. Paper presented at the Thirteenth Conference on Computational Natural Language Learning, Colorado doi:10.3115/1596374.1596381
- Morsy, S. A. (2011). *Recognizing contextual valence shifters in document-level sentiment classification (Master)*. American University in Cairo.
- Mostafa, A. M. (2017). An automatic lexicon with exceptional-negation algorithm for arabic sentiments using supervised classification. *Journal of Theoretical & Applied Information Technology*, 95(15), 3662–3671.
- Pang, B., & Lee, L. (2004). *A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts*. Paper presented at the the 42nd annual meeting on Association for Computational Linguistics, Barcelona. doi:10.3115/1218955.1218990
- Pang, B., Lee, L., & Vaithyanathan, S. (2002). *Thumbs up?: Sentiment classification using machine learning techniques*. Paper presented at the ACL-02 conference on Empirical methods in natural language processing. doi:10.3115/1118693.1118704
- Polanyi, L., & Zaenen, A. (2006). *Contextual valence shifters Computing attitude and affect in text: Theory and applications*. Springer.
- Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2), 267–307. doi:10.1162/COLI_a_00049
- Villalba-Osornio, S. G., Pérez-Celis, J. A., Villasenor-Pineda, L., & Montes-y-Gómez, M. Sentiment analysis for reviews in spanish: Algorithm for negation handling. *Journal on Computer Science and Computer Engineering*, (97), 21-34.
- Wiegand, M., Balahur, A., Roth, B., Klakow, D., & Montoyo, A. (2010). *A survey on the role of negation in sentiment analysis*. Paper presented at the the workshop on negation and speculation in natural language processing, Uppsala.
- Zhang, L., Ferrari, S., & Enjalbert, P. (2012). *Opinion analysis: The effect of negation on polarity and intensity*. Paper presented at the KONVENS, Vienna.
- Zou, B., Zhu, Q., & Zhou, G. (2015). *Negation and speculation identification in chinese language*. Paper presented at the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing. doi:10.3115/v1/P15-1064

Omar Al-Harbi is an assistant professor at the Department of Computer Science at Community College/Jazan University where he has been a faculty member since 2013. He received his PhD degree in Computer Science from Islamic Science University of Malaysia (USIM) in 2013. His current research interests include natural language processing, machine learning, and applications of natural language processing tasks, particularly word sense disambiguation, sentiment analysis, and question answering systems. His current research focuses on deep learning for NLP, including Arabic sentiment analysis.