Sentiment Thesaurus, Synset and Word2Vec Based Improvement in Bigram Model for Classifying Product Reviews

S. Poomagal1 · B. Malar1 · E. M. Ranganayaki1 · K. Deepika1 · G. Dheepak1

Received: 31 August 2020 / Accepted: 2 July 2022
© The Author(s), under exclusive licence to Springer Nature Singapore Pte Ltd 2022

Abstract
Classifying product reviews is one of the tasks in Natural Language Processing by which the sentiment of the reviewer towards a product can be identified. This identification is useful for the growth of the business by increasing the number of satisfied customers through product quality improvement. Bigram models are more popular in performing this classification since it considers the occurrence of two words consecutively in the reviews. In the existing works on bigram models, semantically similar words to the words present in bigrams are not considered. As the reviewers use different words with the same meaning to express their feeling, we proposed improved bigram models in which semantically similar words to the words in bigrams are also used for classifying the reviews. In the proposed models, sentiment polarity thesaurus is constructed by including sentiment words and their synonyms. The combinations of constructed thesaurus, Synset and Word2Vec are used for extracting synonyms for the words in the reviews. Performance of the proposed models is compared with the traditional bigram model and state-of-the-art methods. It is observed from the results that our models are able to achieve better performance than traditional model and recent methods.

Keywords Classification · Synset · Word2Vec · Bigram · Natural Language Processing · Unigram

List of Symbols

| $R$ | Set of training reviews |
|---|---|
| $r_i$ | $i$th review |
| $n$ | Number of reviews |
| $PR$ | Set of positive training reviews |
| $pr_i$ | $i$th positive training review |
| $pn$ | Number of positive training reviews |
| $NR$ | Set of negative training reviews |
| $nr_i$ | $i$th negative training review |
| $nn$ | Number of negative training reviews |
| $pw_{ij}$ | $j$th word in $i$th positive training review |
| $wc_i$ | Number of words in $i$th positive training review |
| $nw_{ij}$ | $j$th word in $i$th negative training review |
| $wnc_i$ | Number of words in $i$th negative training review |
| $plist$ | List of words from positive training reviews |
| $nlist$ | List of words from negative training reviews |
| $pwc$ | Number of words in $plist$ |
| $nwc$ | Number of words in $nlist$ |
| posprob$_{w_iw_j}$ | Probability of $w_i$ followed by $w_j$ in positive training reviews |
| negprob$_{w_iw_j}$ | Probability of $w_i$ followed by $w_j$ in positive training reviews |
| poslist | Set of positive words collected from http://ptrckprr.com/course/ssd/data/positive-words.txt |
| posw$_i$ | $i$th word in poslist |
| pcount | Number of words in poslist |

1 PSG College of Technology, Coimbatore, Tamilnadu, India
The objectives of the proposed method are three fold.

**Introduction**

Sentiment analysis or opinion mining of product reviews understands the sentiment of users for a particular product. This sentiment can be positive or negative based on the quality and features of the product and each review can be assigned to one of these two classes according to the words present in it. This assignment task can be done with the help of classification algorithms and the bigram model.

While considering classification algorithms, the features in the reviews are to be represented in the form of a vector space model with weight assignment for each feature. For assigning weight for each feature, the occurrence of each feature independent of other surrounding features is considered. This is overcome when using the n-gram model in which the classifier model is generated by looking at the occurrence of n features continuously in a review. In the n-gram model, bigrams usually give better results when used for sentiment analysis. Bigram model can further be improved by utilizing semantically similar features of the list of existing features since different reviewers use a variety of similar words to express their sentiments. This can be caught when including synonyms also for classifying the reviews. For including synonyms in the bigrams, thesaurus containing words and their synonyms can be constructed. Since constructing a thesaurus for the entire collection of English words is tough, only positive and negative sentiment words can be considered. But to improve the performance further, synonym for non-sentiment words can also be included. This can be done using Synset and Word2Vec.

This has motivated us to improve bigram model by including semantically similar words using combinations of thesaurus, Synset and Word2Vec. In the proposed thesaurus-based method, thesaurus of positive and negative polarity words is formed and it is used to find similar words for the sentiment-related words in the reviews. For the remaining non-sentiment words, a combination of Synset and Word2Vec dictionaries is used for getting similar words. In the proposed Synset and Word2Vec-based models, semantically similar words are extracted for both sentiment and non-sentiment words from Synset and Word2Vec. Also when finding the probability of test review, instead of considering only the original bigram, probabilities of the combinations of similar words represented by each word in bigrams are taken and class is assigned based on the cumulative probability of those combinations.

The objectives of the proposed method are three fold,
1. It proposes new improved bigram models for classifying reviews as positive or negative.
2. It improves the classification performance by including synonyms of words in the reviews from sentiment thesaurus, NLTK’s Synset and Google’s Word2Vec.
3. Experiments are conducted on different datasets and results are analyzed and reported.

This paper is organized as follows: “Related Work” discusses the existing works on the bigram model. “Problem Statement” states the problem. “Proposed Method” explains the steps in the proposed method. “Experimental Results” states the results produced and “Conclusion” concludes the paper.

Related Work

Product review classification finds the feelings of reviewers by assigning classes for each review as positive or negative. The extracted information tells the quality of the product and it can be used to improve the product. This analysis can also be used to automate the task of assigning the polarity of the future reviews and suggesting the same kind of products to the users.

Many research works are going on in the area of sentiment classification. Pang et al. [1] have utilized the bag of words model for finding the sentiment class of the text content. The class can be positive or negative based on the terms present in the text. Various machine learning algorithms and n-gram are used for the classification of text. Salvetti et al. [2] have executed machine learning algorithms NB and Markov model using hypernym from Wordnet and Part of Speech (PoS) tagging. They concluded that PoS gave better results than Wordnet.

NB model is [3] used for sentiment classification. In this work, a pair of derived features that can be combined are extracted and to improve the accuracy of the result, additional derived features are added. Sahu and Ahuja [4] have proposed a technique in which polarity-based feature selection is done. Once the features are selected, various classification algorithms are used and came to conclusion that the random forest algorithm gave better results when compared to other classification algorithms.

Bodapati et al. [5] did sentence-based sequence modeling and NN-based classification using LSTM. They have converted the words into vectors using Skipgram model of Word2Vec. They compared their work with basic classification algorithms and concluded that LSTM performed better than other techniques. Tripathy et al. [6] compared various machine learning algorithms using unigram, bigram and trigram with various feature selection schemes such as Term Occurrence, Binary term occurrence, Term frequency and TF-IDF. In the case of NB, they concluded that bigram performs better than unigram and trigram.

Vinodhini and Chandrasekaran [7] have proposed a system consisting of 3 models Unigram, Unigram + Bigram, Unigram + Bigram + Trigram and found that 3rd model outperformed others when classified using a hybrid combination of PNN and PCA. Bakliwal et al. [8] proposed a model in which the combination of Unigram, Bigram and Trigram is done and also used POS tagging for this representation. It is found that the combination of all three representations along with POS tag outperformed when classified using MLP.

Acosta et al. [9] have compared two training algorithms using CBOW(Continuous Bag of Words) and SG (SkipGram). The models were tested using both hierarchical Soft-Max and negative sampling. They observed that SG based training model outperformed CBOW based model. Bansal and Srivastava [10] converted every document to a weighted combination of word vectors using TFIDF scheme and discovered that CBOW outperformed SG in various machine learning algorithms especially Random Forest. Mohammed and Fatima [11] analyzed and compared Unigram, Unigram-PoS, Bigram, SkipGram and found the combination of Unigram-PoS + SkipGram had performed better in both MNB and SVM classification.

Awachate et al. [12] introduced a model with the combination of features selected using Chi-square and list of words which is an intersection between the words in the corpuses and in the sentiment word list. The proposed model was found to have a better performance over the conventional representation when classified using Kernlab SVM. Hameed et al. [13] used BoW representation in which feature selection is performed by Information gain using Chi-square method and it was discovered that a combination of Unigram + Bigram with this representation gave better results.

Aspect-based sentiment analysis of movie reviews is done by Thet et al. [14]. They considered that the movie review includes independent clauses that specify different sentiments toward various aspects of a movie. They have used a language-based approach for finding the sentiment of the clauses using the sentiment scores calculated for each word based on the grammatical dependency structure of the clause. Sentiment scores of 32,000 words taken from SentiWordNet have been used by them. Singh et al. [15] used machine learning classifiers namely, Naïve Bayes, J48, BFTree and OneR for optimization of sentiment analysis.

Performance of SVM, Maximum Entropy and Scoring methods when classifying movie reviews have been compared by Tsutsumi et al. [16]. They considered 1400 reviews...
from which half are positive reviews and the remaining half are negative reviews. It is proved in the paper that SVM performs better than the other two techniques.

Word2Vec Convolutional Neural Network (CNN) with random, Skipgram and CBOW models are used [17] to classify news articles and tweets. They proved that skipgram model performs better in the case of tweets and CBOW model works well when the news articles are classified.

Sasikala and Mary [18] proposed a technique to perform sentiment analysis of product reviews by using Deep Learning Modified Neural Network and Improved Adaptive Neuro-Fuzzy Inferences System. The first model is used to classify the review as positive or negative. The second model predicts the demand for a product in the market.

SVM-based product review classification for Indonesian reviews is done with the Word2Vec model as features by Fauzi [19]. He presented the results using binary TF, Raw TF and TFIDF in terms of the measure accuracy.

Poomagal et al. [20] proposed a new Tag_score model for tweets clustering using improved K-means. They used synonyms to reduce the number of words in the collection and the sentiment score of each word to compute the initial centroid for K-means algorithm.

News text data classification is performed by Zhang [21] using a new intelligent model. The intelligent model is constructed for cross-domain text sentiment classification. A domain invariant dictionary is constructed to combine two different methods.

Fei et al. [22] combined improved cross-entropy loss function with the CNN model and LSTM network for cross-domain sentiment classification. They analyzed the influence of each word in improving classification performance.

Souma et al. [23] classified news article based on stock price returns. They utilized a deep learning algorithm for the purpose of classification. They utilized word vectors created from Wikipedia and Gigaword by Glove.

Chandra and Krishna [24] used a deep learning model for analyzing the sentiments of tweets during the rise of COVID-19 in India. They concluded that most of the tweets are optimistic. Miranda et al. [25] have analyzed people’s sentiment on the Indonesian general election using Sentiwordnet and Naïve Bayes algorithm.

**Problem Statement**

Given a set of labeled product reviews, the proposed models aim to assign sentiment for the reviews with a high accuracy rate. This is performed by first constructing bigrams of semantically similar words of the words from each test review using sentiment thesaurus, Synset and Word2Vec and then the reviews are classified into positive or negative based on the probability.

**Proposed Method**

The proposed models classify the reviews into positive or negative by initially finding similar words of words present in test review from the thesaurus, Synset and Word2Vec. Steps in the proposed method are as follows:

1. Preprocessing of reviews
2. Bigram probability matrix formation
3. Classifying a test review
   
   (i) Sentiment thesaurus formation
   (ii) Similar words extraction
   (iii) Bigram with similar words probability calculation
   (iv) Predicting class of a test review

Initially, the training reviews are preprocessed to collect only the words which are useful for performing classification. Then the words from positive reviews and negative reviews are formed as bigram models by computing the probability of each bigram and constructing a probability matrix for each category separately. When classifying a test review, thesaurus, Synset and Word2Vec are used. When using a thesaurus, for each word in the test review, semantically similar words are taken from the thesaurus if the word is a sentiment word, else, similar words are retrieved from Synset or Word2Vec. When using Synset and Word2Vec, semantically similar words are extracted from Synset and Word2Vec only for both sentiment and non-sentiment words. Then these extracted similar words are compared with the list of words in training reviews. The words present in positive/negative training reviews are combined separately to form two different sets for each word of the test review. Based on the probability of newly formed test review bigrams with similar words combination in training reviews, class is assigned for test review.

**Preprocessing of Reviews**

The reviews are preprocessed by removing stop words, punctuations, and other non-alphabetical characters and converting all the characters to lower case. For instance, the review “Exterior of the phone was good” is changed to “exterior phone good”.

Let $R = \{r_i\}_{i=1}^{n}$ be the set of training reviews, $PR = \{pr_i\}_{i=1}^{pm}$ and $NR = \{nr_i\}_{i=1}^{nm}$, represent the collection of positive reviews and negative reviews in the training dataset after preprocessing. Let $pr_i = \{pw_{ij}\}_{j=1}^{wc_i}$ and $nr_i = \{nw_{ij}\}_{j=1}^{wnc_i}$ be the set of words in $i$th positive and negative review,
respectively. Let $w_{c_i}$ and $w_{nc_i}$ be the number of words in $i$th positive and negative review, respectively.

For example, the training dataset contains $n = 8$ reviews represented as $r_1, r_2, …, r_8$. From these 8 reviews, assume that $p_n = 4$ reviews are assigned with positive sentiment and $n_n = 4$ reviews are assigned with negative sentiment. Positive reviews are represented as $\{pr_1, pr_2, pr_3, pr_4\}$ and negative reviews are represented as $\{nr_1, nr_2, nr_3, nr_4\}$.

Each review has a collection of words and let us assume the number of words in 4 positive reviews is given as $w_{c_1} = 3$, $w_{c_2} = 2$, $w_{c_3} = 4$, $w_{c_4} = 2$ and in 4 negative reviews is given as $w_{nc_1} = 2$, $w_{nc_2} = 2$, $w_{nc_3} = 3$, $w_{nc_4} = 2$. So each positive review $pr_i$ is represented as a set of words $\{pw_{i1}, pw_{i2}, …, pw_{iwc_i}\}$. That is, $pr_1$ is represented as $\{pw_{11}, pw_{12}, pw_{13}\}$, $pr_2$ is represented as $\{pw_{21}, pw_{22}\}$, $pr_3$ is represented as $\{pw_{31}, pw_{32}, pw_{33}, pw_{34}\}$, $pr_4$ is represented as $\{pw_{41}, pw_{42}\}$ and so on. Similarly, negative reviews are represented with the sets of words in it.

Another task done during preprocessing is to extract all distinct words from the reviews and to form two lists. One list containing distinct words from positive reviews and another list has words from negative reviews. This is done since these lists are used while forming the list of synonymous words during testing.

Let $plist$ and $nlist$ be two lists containing words from positive reviews and negative reviews, respectively. The respective count values are $pwc$ and $nwc$. The lists are defined in Eqs. (1) and (2):

$$plist = \{pw_{ij} \text{ if } pw_{ij} \text{ is not in } plist \text{ where } 1 \leq i \leq w_{c_j} \text{ for each } j \in 1 \leq j \leq p_n\}$$

(1)

$$nlist = \{nw_{ij} \text{ if } nw_{ij} \text{ is not in } nlist \text{ where } 1 \leq i \leq w_{nc_j} \text{ for each } j \in 1 \leq j \leq n_n\}$$

(2)

**Bigram Probability Matrix Formation**

After preprocessing, the positive and negative training reviews are represented as a matrix containing words present in the reviews as rows and columns. Each entry in the matrix represents the probability with which the word representing the corresponding row can follow the word representing the respective column in the positive/negative reviews collection. Let $posprob_{w_iw_j} = \text{prob}(w_j | w_i)$ where $1 \leq i, j \leq pwc$ and $negprob_{w_iw_j} = \text{prob}(w_j | w_i)$ where $1 \leq i, j \leq nwc$ represent the probability matrix of positive reviews and negative reviews, respectively.

**Classifying a Test Review**

Once the processing of the training dataset is over, test review can be classified into positive or negative by first finding the similar context words of the words present in the test data, finding the probability of all combinations of its bigrams in the training dataset. Finally, based on the logarithmic summation of probabilities of test review bigrams in positive and negative training reviews, class is predicted.

**Sentiment Thesaurus Formation**

For forming thesaurus for sentiment words, a set of positive and negative words are collected from http://ptrckprry.com/course/ssl/data/positive-words.txt and http://ptrckprry.com/course/ssl/data/negative-words.txt [26]. Synonyms for each word in the set are extracted from thesaurus.com. Two major advantages of using web-based thesaurus are that it has a huge vocabulary and it includes many synonyms for any given sentiment word. Also it has synonyms for almost all sentiment words passed to it. Major reason for using this thesaurus in product review classification is due to the fact that the reviewers use words with the high sentiment more often in presenting their view about a product.

The words collected are represented using $poslist$ and $neglist$ as in Eqs. (3) and (4). Equations (5) and (6) show the respective synonyms extracted from the website. Let $possyn_i$ and $negsyn_i$ be the synonym set of $i$th word in positive list and negative list, respectively:

$$poslist = \{posw_i\}_{i=1}^{pcount}$$

(3)

$$neglist = \{negw_i\}_{i=1}^{ncount}$$

(4)

$$possyn_i = \{possynw_{ij}\}_{j=1}^{psyncount}$$

(5)

$$negsyn_i = \{negsynw_{ij}\}_{j=1}^{nsyncount}$$

(6)

**Similar Words Extraction**

All unique words from the test review are collected by applying preprocessing, the extracted words are compared with the constructed thesaurus and if it is a sentiment word and present in it, the respective set of similar words are collected. If the word is a non-sentiment word, then it is sent to Syn-set of Wordnet or Google’s Word2Vec to find semantically similar words.
similar words for them. Also, for both sentiment and non-sentiment words, Synset and Word2Vec are used.

Once the words are retrieved, they are compared with positive training words and negative training words to construct two collections for each word. The first collection with all similar words of the word which are also present in positive training reviews and the second one with all similar words occurring in negative training reviews. This process is mathematically defined as follows.

Let \( W = \{w_i\}_{i=1}^m \) represent the set of unique words from the test review. Let \( \text{psyn}_i \) and \( \text{nsyn}_i \) be the set of similar words of \( i \)th word in a positive thesaurus or Synset and negative thesaurus or Synset, respectively. Let \( p\text{Word}2\text{Vec}_i \) and \( n\text{Word}2\text{Vec}_i \) be the set of similar context words of \( i \)th word in positive thesaurus or Word2Vec and negative thesaurus or Word2Vec, respectively. They are described in Eqs. (7)–(10):

\[
\text{psyn}_i = \left\{ \begin{array}{ll}
\text{possyn}_i & \text{if } \text{pos}_w = w_i \text{ and } 1 \leq j \leq p\text{count} \\
\text{Synset}(w_i) & \text{Otherwise}
\end{array} \right.,
\]

\[
\text{nsyn}_i = \left\{ \begin{array}{ll}
\text{negsyn}_i & \text{if } \text{neg}_w = w_i \text{ and } 1 \leq j \leq n\text{count} \\
\text{Synset}(w_i) & \text{Otherwise}
\end{array} \right.,
\]

\[
p\text{Word}2\text{Vec}_i = \left\{ \begin{array}{ll}
\text{possyn}_i & \text{if } \text{pos}_w = w_i \text{ and } 1 \leq j \leq p\text{count} \\
\text{Word}2\text{Vec\_sim}(w_i) & \text{Otherwise}
\end{array} \right.,
\]

\[
n\text{Word}2\text{Vec}_i = \left\{ \begin{array}{ll}
\text{negsyn}_i & \text{if } \text{neg}_w = w_i \text{ and } 1 \leq j \leq n\text{count} \\
\text{Word}2\text{Vec\_sim}(w_i) & \text{Otherwise}
\end{array} \right..
\]

When using synonyms of sentiment words only from Synset or Word2Vec, Eqs. (7)–(10) are changed to include only \( \text{synset}(w_i) \) if \( \text{pos}_w = w_i \) and \( 1 \leq j \leq p\text{count} \), \( \text{synset}(w_i) \) if \( \text{neg}_w = w_i \) and \( 1 \leq j \leq n\text{count} \), \( \text{Word}2\text{Vec\_sim}(w_i) \) if \( \text{pos}_w = w_i \) and \( 1 \leq j \leq p\text{count} \), and \( \text{Word}2\text{Vec\_sim}(w_i) \) if \( \text{neg}_w = w_i \) and \( 1 \leq j \leq n\text{count} \), respectively.

Let \( p\text{vt}_w \) and \( n\text{vt}_w \) represent words present in \( p\text{syn}_i \) and \( n\text{syn}_i \), which are also available in positive training reviews and negative training reviews, respectively. Initially, \( \cup_{1 \leq m \leq p\text{vt}_w} n\text{vt}_w = n\text{vt}_w = w_i \) and the process is defined in Eqs. (11) and (12):

\[
p\text{vt}_w = \{ p\text{syn}_i \text{ if } p\text{syn}_i \in \text{plist} \text{ and } 1 \leq j \leq p\text{c}_i \}. \tag{11}
\]

\[
n\text{vt}_w = \{ n\text{syn}_i \text{ if } n\text{syn}_i \in \text{ntlist} \text{ and } 1 \leq j \leq n\text{c}_i \}. \tag{12}
\]

In the same way, words in \( p\text{Word}2\text{Vec}_i \) or \( n\text{Word}2\text{Vec}_i \) which are also occurring in positive and negative training reviews are represented as \( p\text{vt}_w \) and \( n\text{vt}_w \), respectively. Initially, \( \cup_{1 \leq m \leq p\text{vt}_w} n\text{vt}_w = n\text{vt}_w = w_i \) and the above process is described in Eqs. (13) and (14):

\[
p\text{vt}_w = \{ p\text{Word}2\text{Vec}_i \text{ if } p\text{Word}2\text{Vec}_i \in \text{plist} \text{ and } 1 \leq j \leq p\text{c}_i \}. \tag{13}
\]

\[
n\text{vt}_w = \{ n\text{Word}2\text{Vec}_i \text{ if } n\text{Word}2\text{Vec}_i \in \text{ntlist} \text{ and } 1 \leq j \leq n\text{c}_i \}. \tag{14}
\]

where \( p\text{c}_i, n\text{c}_i, p\text{c}_i \) and \( n\text{c}_i \) represent the number of words in \( p\text{syn}_i \), \( n\text{syn}_i \), \( p\text{Word}2\text{Vec}_i \) and \( n\text{Word}2\text{Vec}_i \), respectively.

**Bigrams with Similar Words Probability Calculation**

Once the sets are formed for each word in the test review, the combinations of the similar words in positive/negative sets are used to form bigrams and the probability of each formed bigram in positive/negative training datasets is found. Finally, the summation of all positive/negative logarithmic probabilities is calculated and the respective class is assigned to the test review. Equations (15) and (16) explain this process for Synset and thesaurus-based model.

\[
P(\hat{y}_1 | w_1 \ldots w_m) = \forall w_i \in p\text{vt}_w \forall w_i \in p\text{vt}_{w+1} \sum \log \left( \text{posprob}_w \right)
\]

where \( 1 \leq x \leq m - 1 \), \tag{15}

\[
P(\hat{y}_2 | w_1 \ldots w_m) = \forall w_i \in n\text{vt}_w \forall w_i \in n\text{vt}_{w+1} \sum \log \left( \text{negprob}_w \right)
\]

where \( 1 \leq x \leq m - 1 \). \tag{16}

In the same way, the probability of test review belonging to each category is calculated in Word2Vec and thesaurus-based model as mentioned in Eqs. (17) and (18):
Predicting Class of a Test Review

Test review sentiment is predicted based on the summation of logarithmic probabilities of test bigrams in positive/negative training reviews. This prediction process is given in Eq. (19):

\[
\hat{y} = \text{argmax}_{k=1,2} P(\hat{y}_k|w_1 \ldots w_m).
\]

Experimental Results

To evaluate the performance of the proposed bigram improvement methods, 10 product review datasets from Amazon.com are considered. These datasets contain reviews by customers of various products which express their sentiment of them about the product. Each dataset has 1000 positive and 1000 negative reviews.

Experimental Setup

Experiments are conducted by predicting the class of test reviews based on probability values calculated using the scenarios presented in Table 1. Methods from M2–M6 consider similar words from only one resource in each case whereas the remaining methods M7–M13 consider similar words from combinations of all three resources.

Proposed models are also compared with state-of-the-art methods such as unigram with TFIDF [6] and bigram using TFIDF [6] and Word2Vec-based CNN [17]. All these comparisons are performed using the measures accuracy, precision, recall, f-measure, completeness and false alarm rate.

For calculating these measures, the terms True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) are used. They are defined in Table 2.

Accuracy describes the percentage of test reviews that are classified correctly. Precision is the number of true positive results divided by the number of actual positive results. Recall is the number of true positive results divided by the number of all reviews that should have been identified as positive. F-measure is the harmonic mean of precision and recall. Completeness measures the percentage of actual positive reviews predicted as positive in the set of reviews. False alarm rate tells the percentage of actual positive reviews predicted as negative reviews in the total number of actual negative reviews. Mathematical definition of these metrics is presented in Eqs. (20)–(25):

\[
\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \times 100.
\]

### Table 1 Scenarios

| Scenario | Description |
|----------|-------------|
| M1       | Traditional bigram model |
| M2       | Similar words extraction from only Synset |
| M3       | Similar words extraction from only Word2Vec |
| M4       | Similar words extraction from thesaurus only for sentiment words |
| M5       | Similar words extraction from Synset only for sentiment words |
| M6       | Similar words extraction from Word2Vec only for sentiment words |
| M7       | Similar words extraction from thesaurus and Synset for sentiment words |
| M8       | Similar words extraction from thesaurus and Word2Vec for sentiment words |
| M9       | Similar words extraction from Synset and Word2Vec for sentiment words |
| M10      | Similar words extraction from thesaurus, Synset and Word2Vec for sentiment words |
| M11      | Similar words extraction from thesaurus for sentiment words and from Synset for non-sentiment words |
| M12      | Similar words extraction from thesaurus for sentiment words and from Word2Vec for non-sentiment words |
| M13      | Similar words extraction from thesaurus for sentiment words and Synset and Word2Vec for non-sentiment words |

### Table 2 Actual and predicted class

| Predicted | Actual |
|-----------|--------|
|           | Positive class | Negative class |
| Positive class | TP     | FP     |
| Negative class  | FN     | TN     |
Precision = \frac{TP}{TP + FP}, \tag{21}

\text{Recall} = \frac{TP}{TP + FN}, \tag{22}

F\text{-measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{23}

\text{Completeness} = \frac{TP}{TP + FP + TN + FN}. \tag{24}

\text{False alarm rate} = \frac{FN}{TN + FP}. \tag{25}

Sample Words from Thesaurus, Synset and Word2Vec

Positive and negative sentiment words are downloaded [18] and their respective synonyms are extracted and thesaurus is constructed. Table 3 shows the sample set of sentiment words along with their synonyms.

Wordnet.synsets() function in NLTK library and model. wv.most_similar() function in the Google Word2Vec model are used for extracting similar words from Synset

Sample sentiment words in thesaurus

| Words    | Synonyms                                                               |
|----------|------------------------------------------------------------------------|
| Beautiful | 'alluring', 'appealing', 'charming', 'cute', 'dazzling', 'delicate', 'delightful', 'elegant', 'exquisite', 'fascinating', 'fine', 'good-looking', 'gorgeous', 'graceful', 'grand', 'handsome', 'lovely', 'magnificent', 'marvelous', 'pleasing', 'pretty', 'splendid', 'stunning', 'superb', 'wonderful', 'admirable', 'angelic', 'beauteous', 'bewitching', 'classy', 'comely', 'divine', 'enticing', 'excellent', 'fair', 'foxy', 'ideal', 'nice', 'pulchritudinous', 'radiant', 'ravishing', 'refined', 'resplendent', 'shapely', 'sightly', 'statuesque', 'sublime', 'symmetrical', 'taking', 'well-formed' |
| Better    | 'exceptional', 'improved', 'superior', 'choice', 'exceeding', 'fitter', 'preferred', 'sophisticated', 'surpassing', 'bigger', 'finer', 'greater', 'larger', 'more desirable', 'more suitable', 'more valuable', 'preferable', 'prominent' |
| Disagree  | 'clash', 'contradict', 'differ', 'dissent', 'diverge', 'conflict', 'counter', 'depart', 'deviate', 'discord', 'disharmonize', 'vary', 'war', 'be dissimilar' |
| Fail      | 'decline', 'fall', 'abot', 'backslide', 'blunder', 'deteriorate', 'fizzle', 'flop', 'flounder', 'fold', 'founder', 'miscarry', 'miss', 'slip', 'be defeated', 'be found lacking', 'be ruined', 'come to nothing', 'fall short', 'go astray', 'go down swinging', 'go up in smoke', 'hit bottom', 'lose control', 'lose status', 'miss the boat', 'run arogrand' |

Semantically similar retrieved using Synset and Word2Vec

| Words    | Synset | Word2Vec |
|----------|--------|----------|
| Picture  | 'picture', 'image', 'impression', 'scene', 'movie', 'film', 'pic', 'video', 'photograph', 'photo', 'fancy', 'see', 'figure', 'show' | 'pictures', 'photograph', 'photo', 'photos', 'images', 'image' |
| Guess    | 'guess', 'guessing', 'think', 'suppose', 'imagine', 'pretend' | 'suppose', 'think', 'yeah', 'maybe', 'probably', 'anyway', 'know', 'hey' |
| Place    | 'place', 'position', 'shoes', 'home', 'post', 'office', 'situation', 'space', 'put', 'set', 'lay', 'rate', 'range', 'order', 'grade', 'locate', 'site', 'point', 'send' | 'places', 'placed', 'finish' |
| Thing    | 'thing', 'matter' | 'things', 'something', 'stuff', 'really', 'think', 'aspect', 'reason', 'kind' |
| Sleep    | 'sleep', 'slumber', 'sopor', 'nap', 'rest', 'eternal_rest', 'eternal_sleep', 'quietus', 'kip' | 'sleeping', 'restful_sleep', 'restorative_sleep', 'slept', 'nap', 'wakings', 'naps', 'fitful_sleep', 'sleep', 'doze' |
| Happy    | 'happy', 'felicitous', 'glad', 'well-chosen' | 'glad', 'pleased', 'ecstatic', 'overjoyed', 'thrilled', 'satisfied', 'delighted', 'disappointed', 'excited' |
| Sad      | 'sad', 'deplorable', 'distressing', 'lamentable', 'ptiifful', 'sorry' | 'saddening', 'Sad', 'saddened', 'heartbreaking', 'disheartening', 'sadens_me', 'distressing', 'reminders_bobbing' |
| Good     | 'good', 'goodness', 'commodity', 'trade_good', 'full', 'estimable', 'honorable', 'respectable', 'beneficial', 'just', 'upright', 'adept', 'expert', 'practiced', 'proficient', 'skilful', 'skilful', 'dear', 'near', 'dependable', 'safe', 'secure', 'right', 'ripe', 'well', 'effective', 'in_effect', 'in_force', 'serious', 'sound', 'salutary', 'honest', 'undecomposed', 'unspoiled', 'unspoil', 'thoroughly', 'soundly' | 'great', 'terriific', 'decent', 'nice', 'excellent', 'fantastic', 'better', 'solid', 'lousy' |
| Bad      | 'bad', 'badness', 'big', 'tough', 'spoiled', 'spoil', 'regretful', 'sorry', 'uncollectible', 'risk', 'high-risk', 'speculative', 'unfit', 'unsound', 'forged', 'defective', 'badly' | 'good', 'terrible', 'horrible', 'Bad', 'lousy', 'crummy', 'horrid', 'awful', 'dreadful', 'horrendous' |
and Word2Vec. Table 4 shows the sample words in the reviews and their respective Synset words and Word2Vec words retrieved.

Once the semantically similar words are extracted, they can be used to put out the combinations in extending the set of test review bigrams. Table 5 shows the sample bigram and the respective combination using one or both of its similar words.

Figure 1 shows the number of sentiment words for which similar words are extracted from thesaurus, number of non-sentiment words for which similar words are extracted from Synset or Word2Vec.

Table 5 Sample bigram combinations using similar words

| Existing bigram     | Sample new bigrams                                      |
|---------------------|---------------------------------------------------------|
| ('phone', 'good')   | ('phone', 'beneficial')                                 |
|                     | ('phone', 'sound')                                      |
|                     | ('phone', 'effective')                                  |
| ('telephone', 'good')| ('telephone', 'beneficial')                            |
|                     | ('telephone', 'sound')                                  |
| ('headphone', 'good')| ('headphone', 'beneficial')                             |
|                     | ('headphone', 'sound')                                  |
| ('earphone', 'good') | ('earphone', 'beneficial')                              |
|                     | ('earphone', 'sound')                                   |
|                     | ('earphone', 'effective')                               |
| ('telephone', 'sound')| ('telephone', 'good')                                  |
| ('headphone', 'sound')| ('headphone', 'good')                                  |

('good', 'price')

| ('beneficial', 'cost') | ('beneficial', 'price') |
| ('sound', 'cost')      | ('sound', 'price')     |
| ('effective', 'cost')  | ('effective', 'price') |

('price', 'paid,')

| ('cost', 'paid,')      | ('cost', 'give,')     |
| ('cost', 'pay,')       | ('cost', 'devote,')   |
| ('cost', 'bear,')      | ('price', 'give,')    |
| ('price', 'pay,')      | ('price', 'devote,')  |
| ('price', 'bear,')     |                         |

('battery', 'lasts')

| ('battery', 'survive') | ('barrage', 'survive') |
| ('barrage', 'lasts')   | ('battery', 'live')    |
| ('barrage', 'live')    | ('battery', 'live')    |

('camera', 'decent')

| ('camera', 'nice')    | ('camera', 'adequate') |
| ('camera', 'enough')  | ('camera', 'properly') |
| ('camera', 'right')   |                         |

and Word2Vec. Table 4 shows the sample words in the reviews and their respective Synset words and Word2Vec words retrieved.

Once the semantically similar words are extracted, they can be used to put out the combinations in extending the set of test review bigrams. Table 5 shows the sample bigram and the respective combination using one or both of its similar words.

Figure 1 shows the number of sentiment words for which similar words are extracted from thesaurus, number of non-sentiment words for which similar words are extracted from Synset or Word2Vec.

Table 6 Number of words for which similar words retrieved

| Dataset   | Total number of words | Thesaurus + Synset | Thesaurus + Word2Vec |
|-----------|-----------------------|--------------------|----------------------|
| Apparel   | 5924                  | 1306               | 1485                 |
| Books     | 14,576                | 2993               | 4666                 |
| DVD       | 17,451                | 3180               | 5136                 |
| Electronics | 11,628              | 2070               | 3146                 |
| Health    | 8274                  | 1574               | 2548                 |
| Kitchen   | 8199                  | 1703               | 2525                 |
| Music     | 12,067                | 2185               | 3745                 |
| Sports    | 9818                  | 1834               | 3003                 |
| Toys      | 9416                  | 1692               | 2752                 |
| Video     | 13,970                | 2593               | 4456                 |

When analyzing the data in Table 5, Google Word2Vec has more number of words so that it returns words for many words present in the reviews. Though it contains many words, the result comparison shows that Synset is better than Word2Vec since Word2Vec gives many useless words as it has a huge vocabulary.

Comparative Study

For comparing the performance of proposed models with the traditional bigram model, test reviews are classified, metrics are calculated and presented. Table 8 shows the accuracy produced by M1–M6 for all 10 datasets. When
analyzing the results, similar words inclusion (M2–M6) have produced better results when compared to the traditional bigram model (M1). Among the methods M2–M6, methods which use Synset (M2, M4 and M5) perform better in 80% of the cases. Though Word2Vec returned similar words for more number of words, performance of it is weaker than Synset. The reason is that it returns many meaningless synonyms as its vocabulary size of it is large. Table 7 lists the words returned by Word2Vec which are not synonyms of the input word.

The combinations of all three resources were also tried and the results are projected in Table 9. It shows that using thesaurus for sentiment words and Synset for non-sentiment words (M11) gave better results for 70% of the cases compared to other combinations. When comparing all the methods M1 to M13, methods which utilizes Synset have produced high accuracy for 90% of the datasets.

Overall, thesaurus for sentiment words and Synset for non-sentiment words improve the performance of classification significantly when considering accuracy measures (Table 10).

Precision, recall and F-measure produced by M1–M13 and analysis of the results are presented in Tables 11, 12, 13, 14, 15, 16 and 17. When comparing F-measure of M1 to M6, Synset-based methods (M2, M4 and M5) succeed over other methods for 90% of the cases. This has shown that Synset is better when compared to Word2Vec methods. The comparison of M7–M13 is presented in Table 16 which presents that Synset-based methods are better compared to other methods in 90% of the cases.

As a consolidation, proposed semantic similarity inclusion methods outperform the traditional bigram model. The reason for this improvement is that the reviewers use different words with the same meaning for presenting their feeling over a product. In this work, all the words and the

---

**Table 7** Words and sample meaningless synonyms returned by Word2Vec

| Word     | Useless words returned by Word2Vec |
|----------|-----------------------------------|
| Guess    | “hey”, “know”                      |
| Thing    | “really”, “think”, “kind”          |
| See      | “expect”, “imagine”                |
| Came     | “got”, “gave”, “ran”               |
| Supposed | “going”, “trying”, “wanted”, “not”, “want” |

---

**Table 8** Accuracy of M1–M6

| Dataset  | Accuracy | Winner |
|----------|----------|--------|
|          | M1   | M2   | M3   | M4   | M5   | M6   |        |
| Apparel  | 70.5 | 75   | 74   | 71   | 72.5 | 73   | M2 (75%) |
| Books    | 70   | 74.5 | 75.5 | 76   | 75.5 | 73   | M4 (76%) |
| DVD      | 66   | 74   | 68.5 | 70.5 | 74.5 | 67   | M5 (74.5%) |
| Electronics | 75  | 79   | 74   | 76   | 78.5 | 73   | M2 (79%) |
| Health   | 74   | 79   | 77   | 77   | 77.5 | 75.5 | M2 (79%) |
| Kitchen  | 72   | 77.5 | 76   | 73.5 | 74   | 71.5 | M2 (77.5%) |
| Music    | 65.5 | 72   | 70   | 68.5 | 71.5 | 67   | M2 (72%) |
| Sports   | 68   | 73.5 | 74   | 73   | 72.5 | 67   | M3 (74%) |
| Toys     | 68   | 67   | 71.5 | 70.5 | 69   | 70.5 | M3 (71.5%) |
| Video    | 70.5 | 78.5 | 73.5 | 74   | 75.5 | 71   | M2 (78.5%) |

---

**Table 9** Accuracy of M7–M13

| Dataset  | Accuracy | Winner |
|----------|----------|--------|
|          | M7   | M8   | M9   | M10  | M11  | M12  | M13  |        |
| Apparel  | 69   | 69   | 68   | 69   | 79   | 77   | 70.5 | M11 (79%) |
| Books    | 78.5 | 78   | 79.5 | 80   | 76   | 74   | 74   | M10 (80%) |
| DVD      | 75   | 77.5 | 76.5 | 76   | 74.5 | 70   | 70   | M8 (77.5%) |
| Electronics | 75  | 75.5 | 72.5 | 72   | 79   | 76   | 67.5 | M11 (79%) |
| Health   | 76.5 | 79   | 77   | 76   | 80   | 76.5 | 70.5 | M11 (80%) |
| Kitchen  | 74   | 72.5 | 77   | 73.5 | 80.5 | 78.5 | 67.5 | M11 (80.5%) |
| Music    | 65.5 | 66   | 68   | 64.5 | 73.5 | 71   | 69   | M11 (73.5%) |
| Sports   | 69.5 | 69.5 | 71   | 69.5 | 74   | 72   | 71.5 | M11 (74%) |
| Toys     | 74   | 73   | 72   | 72.5 | 69.5 | 73.5 | 69   | M7 (74%) |
| Video    | 73.5 | 74.5 | 75.5 | 74.5 | 78   | 73.5 | 73   | M11 (78%) |
combinations of semantically similar words of the words are taken for predicting the class of the given test review. This covers almost all similar combinations for bigrams in the test reviews.

Table 18, 19 and 20 show the completeness score produced by all the methods. From the results, it is observed that for all the datasets, Synset-based methods are superior to other methods. Table 21, 22 and 23 show the false alarm rate which actually needs to be decreased since the formula considers the count of wrongly classified reviews in the numerator. As the results show, Synset-based methods (M2, M4 and M5) are better than both traditional bigram model (M1) and Word2Vec-based methods (M3 and M6) for all datasets. The performance comparison of methods M7–M13 using completeness and false alarm rate is projected in Tables 19 and 22. In terms of both completeness and false alarm rate, Synset-based methods are producing better results.

When we consolidate the results presented above, Synset-based methods perform well in most of the cases. Word2Vec-based methods are also performing equally with a few Synset-based methods but some lag in the values produced. The reason behind this is that though Word2Vec has a large vocabulary, it returns less similar words for many words sent to it. But in case of Synset, even though the vocabulary size is small, it returns more perfect similar words for the given review words. Overall, using Synset with and without

| Dataset     | Winner from Table 7 | Winner from Table 8 | Overall winner |
|-------------|---------------------|---------------------|----------------|
| Apparel     | M2 (75%)            | M11 (79%)           | M11            |
| Books       | M4 (76%)            | M10 (80%)           | M10            |
| DVD         | M5 (74.5%)          | M8 (77.5%)          | M8             |
| Electronics | M2 (79%)            | M11 (79%)           | M2 and M11     |
| Health      | M2 (79%)            | M11 (80%)           | M11            |
| Kitchen     | M2 (77.5%)          | M11 (80.5%)         | M11            |
| Music       | M2 (72%)            | M11 (73.5%)         | M11            |
| Sports      | M3 (74%)            | M11 (74%)           | M3 and M11     |
| Toys        | M3 (71.5%)          | M7 (74%)            | M7             |
| Video       | M2 (78.5%)          | M11 (78%)           | M2             |

| Dataset     | Precision          | Winner          |
|-------------|--------------------|----------------|
| M1          | M2                 | M2 (0.727273)  |
| M2          | M2                 | M2 (0.727273)  |
| M3          | M2                 | M2 (0.727273)  |
| M4          | M2                 | M2 (0.727273)  |
| M5          | M2                 | M2 (0.727273)  |
| M6          | M2                 | M2 (0.727273)  |

| Dataset     | Precision          | Winner          |
|-------------|--------------------|----------------|
| M7          | M8                 | M12 (0.84285714) |
| M8          | M9                 | M12 (0.84285714) |
| M9          | M10                | M12 (0.84285714) |
| M10         | M11                | M12 (0.84285714) |
| M11         | M12                | M12 (0.84285714) |
| M12         | M13                | M12 (0.84285714) |
| M13         |                    | M12 (0.84285714) |

| Dataset     | Precision          | Winner          |
|-------------|--------------------|----------------|
| M7          | M8                 | M12 (0.84285714) |
| M8          | M9                 | M12 (0.84285714) |
| M9          | M10                | M12 (0.84285714) |
| M10         | M11                | M12 (0.84285714) |
| M11         | M12                | M12 (0.84285714) |
| M12         | M13                | M12 (0.84285714) |
| M13         |                    | M12 (0.84285714) |
thesaurus outperforms the conventional bigram model (M1), Word2Vec with and without thesaurus.

**Comparison with State of the Art Methods**

Effectiveness of proposed Synset and Word2Vec with and without thesaurus is also compared with unigram using TFIDF [6] and bigram using TFIDF [6] and Word2Vec CNN with random vector generation [17], Skipgram [17] and CBOW [17]. In [6], they used Naïve Bayes (NB) classification algorithm after computing TFIDF values for unigrams and bigrams. We implemented the same technique and executed it for 10 datasets considered in this work.

| Dataset   | Recall |  |  |  |  |  | M4 and M5 (0.83) |
|-----------|--------|---|---|---|---|---|----------------|
| Apparel   | 0.73   | 0.8 | 0.79 | 0.83 | 0.83 | 0.79 | M4 and M5 (0.83) |
| Books     | 0.6    | 0.73 | 0.64 | 0.71 | 0.71 | 0.6 | M2 (0.73) |
| DVD       | 0.51   | 0.68 | 0.61 | 0.59 | 0.69 | 0.49 | M5 (0.69) |
| Electronics | 0.67 | 0.78 | 0.7 | 0.75 | 0.8 | 0.67 | M5 (0.8) |
| Health    | 0.62   | 0.76 | 0.73 | 0.74 | 0.78 | 0.68 | M5 (0.78) |
| Kitchen   | 0.72   | 0.82 | 0.79 | 0.78 | 0.82 | 0.73 | M2 and M5 (0.82) |
| Music     | 0.66   | 0.82 | 0.77 | 0.71 | 0.83 | 0.65 | M5 (0.83) |
| Sports    | 0.69   | 0.83 | 0.8 | 0.82 | 0.89 | 0.77 | M5 (0.89) |
| Toys      | 0.65   | 0.67 | 0.73 | 0.72 | 0.75 | 0.7 | M5 (0.75) |
| Video     | 0.69   | 0.83 | 0.76 | 0.77 | 0.83 | 0.69 | M2 and M5 (0.83) |

| Dataset   | Recall |  |  |  |  |  | M10 (0.89) |
|-----------|--------|---|---|---|---|---|----------------|
| Apparel   | 0.88   | 0.88 | 0.85 | 0.89 | 0.83 | 0.73 | 0.79 | M10 (0.89) |
| Books     | 0.91   | 0.87 | 0.91 | 0.94 | 0.7 | 0.59 | 0.65 | M10 (0.94) |
| DVD       | 0.85   | 0.83 | 0.86 | 0.89 | 0.64 | 0.52 | 0.64 | M10 (0.89) |
| Electronics | 0.88 | 0.86 | 0.86 | 0.89 | 0.74 | 0.66 | 0.6 | M10 (0.89) |
| Health    | 0.87   | 0.88 | 0.87 | 0.89 | 0.74 | 0.63 | 0.65 | M10 (0.89) |
| Kitchen   | 0.93   | 0.91 | 0.93 | 0.94 | 0.8 | 0.73 | 0.67 | M10 (0.94) |
| Music     | 0.91   | 0.88 | 0.89 | 0.91 | 0.8 | 0.66 | 0.74 | M7 and M10 (0.91) |
| Sports    | 0.87   | 0.87 | 0.89 | 0.89 | 0.82 | 0.65 | 0.81 | M9 and M10 (0.89) |
| Toys      | 0.87   | 0.83 | 0.87 | 0.88 | 0.69 | 0.64 | 0.7 | M10 (0.88) |
| Video     | 0.87   | 0.86 | 0.88 | 0.89 | 0.8 | 0.64 | 0.79 | M10 (0.89) |

| Dataset   | F-measure |  |  |  |  |  | M2 (0.761904762) |
|-----------|-----------|---|---|---|---|---|----------------|
| Apparel   | 0.712195122 | 0.761904762 | 0.752380952 | 0.741071429 | 0.751131222 | 0.74528302 | M2 (0.761904762) |
| Books     | 0.666666667 | 0.741116751 | 0.723163842 | 0.743455497 | 0.743455497 | 0.666666667 | M4 (0.747368421) |
| DVD       | 0.6 | 0.723404255 | 0.659459459 | 0.666666667 | 0.73015873 | 0.59756098 | M5 (0.73015873) |
| Electronics | 0.72826087 | 0.787878788 | 0.729166667 | 0.757575758 | 0.78817734 | 0.71276596 | M5 (0.78817734) |
| Health    | 0.704545455 | 0.783505155 | 0.760416667 | 0.762886598 | 0.776194903 | 0.73513514 | M2 (0.73505155) |
| Kitchen   | 0.72 | 0.784688995 | 0.766990291 | 0.746411483 | 0.759259259 | 0.71921182 | M2 (0.726889995) |
| Music     | 0.656716418 | 0.745454545 | 0.719621688 | 0.692682927 | 0.744394619 | 0.66326531 | M2 (0.745454545) |
| Sports    | 0.683168317 | 0.757990868 | 0.754716981 | 0.752293578 | 0.763948498 | 0.7 | M2 (0.757990868) |
| Toys      | 0.670103093 | 0.67 | 0.719211823 | 0.709359606 | 0.70754717 | 0.70351759 | M3 (0.719211823) |
| Video     | 0.700507614 | 0.794258373 | 0.741463415 | 0.747572816 | 0.772093023 | 0.70408163 | M2 (0.794258373) |
In Word2Vec CNN [17], CNN algorithm is used for classifying the news articles and tweets. The vectors are generated using a random method, skipgram method and CBOW method. In this paper, CNN is executed with all three vector formation techniques for product reviews and the results are projected. Figures 2, 3, 4, 5, 6 and 7 show the comparison of Synset and Word2Vec with and without a thesaurus (M2 to M13) with NB [6] using various measures.

The comparison charts show that for all the cases, bigram model with TFIDF and NB is better than the unigram model with TFIDF and NB. When comparing proposed Synset and Word2Vec using a thesaurus and without using a thesaurus (M2–M13) with this state-of-the-art method, Synset-based models are better in 90% of the cases.

### Table 16 F-measure of M7–M13

| Dataset   | F-measure M7 | F-measure M8 | F-measure M9 | F-measure M10 | F-measure M11 | F-measure M12 | F-measure M13 | Winner       |
|-----------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Apparel   | 0.7394958    | 0.7394958    | 0.72649573   | 0.74166667   | 0.79807692   | 0.76041667   | 0.7281106    | M11 (0.798076923) |
| Books     | 0.80888889   | 0.79816514   | 0.8161435    | 0.8245614    | 0.74468085   | 0.69411764   | 0.71428571   | M10 (0.8245614)   |
| DVD       | 0.77272727   | 0.78672986   | 0.78538813   | 0.78761062   | 0.71508379   | 0.63414634   | 0.68085106   | M10 (0.78761062) |
| Electronics | 0.77876106  | 0.77828054   | 0.75770925   | 0.76068376   | 0.77894736   | 0.73333333   | 0.64864865   | M11 (0.778947368) |
| Health    | 0.78733032   | 0.80733945   | 0.79090909   | 0.78761062   | 0.78723404   | 0.72832369   | 0.68738069   | M8 (0.80733945)   |
| Kitchen   | 0.78151261   | 0.76793249   | 0.80172414   | 0.78082999   | 0.80402010   | 0.77248677   | 0.67336683   | M11 (0.804020101) |
| Music     | 0.7250996    | 0.72131148   | 0.73553719   | 0.71936759   | 0.75117309   | 0.69473684   | 0.7047619    | M11 (0.751173709) |
| Sports    | 0.74042553   | 0.74042553   | 0.75423729   | 0.74476987   | 0.75925925   | 0.69892473   | 0.73972603   | M11 (0.759259259) |
| Toys      | 0.7699115    | 0.75454545   | 0.7562174    | 0.76190476   | 0.69346733   | 0.70718232   | 0.69306931   | M7 (0.7699115)    |
| Video     | 0.76651982   | 0.77130045   | 0.78222222   | 0.77729258   | 0.78431372   | 0.70718232   | 0.74528302   | M11 (0.784313725) |

### Table 17 Analysis of Tables 15 and 16

| Dataset   | Winner from Table 15 | Winner from Table 16 | Overall winner |
|-----------|-----------------------|-----------------------|---------------|
| Apparel   | M2 (0.761904762)      | M11 (0.798076923)     | M11           |
| Books     | M4 (0.747368421)      | M10 (0.8245614)       | M10           |
| DVD       | M5 (0.73015873)       | M10 (0.78761062)      | M10           |
| Electronics | M5 (0.78817734)     | M11 (0.778947368)     | M5            |
| Health    | M2 (0.783501555)      | M8 (0.80733945)       | M8            |
| Kitchen   | M2 (0.784688995)      | M11 (0.804020101)     | M11           |
| Music     | M2 (0.745454545)      | M11 (0.751173709)     | M11           |
| Sports    | M2 (0.757990868)      | M11 (0.759259259)     | M11           |
| Toys      | M3 (0.719211823)      | M7 (0.7699115)        | M7            |
| Video     | M2 (0.794258373)      | M11 (0.784313725)     | M2            |

### Table 18 Completeness of M1–M6

| Dataset   | Completeness M1 | Completeness M2 | Completeness M3 | Completeness M4 | Completeness M5 | Completeness M6 | Winner       |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|--------------|
| Apparel   | 0.365           | 0.4             | 0.395           | 0.415           | 0.415           | 0.395           | M4 and M5 (0.415) |
| Books     | 0.3             | 0.365           | 0.32            | 0.355           | 0.355           | 0.3             | M2 (0.365)   |
| DVD       | 0.255           | 0.34            | 0.305           | 0.295           | 0.345           | 0.245           | M5 (0.345)   |
| Electronics | 0.335           | 0.39            | 0.35            | 0.375           | 0.3            | 0.4             | M5 (0.4)     |
| Health    | 0.31            | 0.38            | 0.365           | 0.365           | 0.37            | 0.39            | M5 (0.39)    |
| Kitchen   | 0.36            | 0.41            | 0.395           | 0.39            | 0.41            | 0.365           | M2 and M5 (0.41) |
| Music     | 0.33            | 0.41            | 0.385           | 0.355           | 0.415           | 0.325           | M5 (0.415)   |
| Sports    | 0.345           | 0.415           | 0.4             | 0.41            | 0.445           | 0.385           | M5 (0.445)   |
| Toys      | 0.325           | 0.335           | 0.365           | 0.36            | 0.375           | 0.35            | M5 (0.375)   |
| Video     | 0.345           | 0.415           | 0.38            | 0.385           | 0.415           | 0.345           | M2 and M5 (0.415) |
For comparing proposed models with CNN [17], CNN algorithm is executed 5 times and the result of the best run is taken and compared with the proposed techniques. Figure 8 projects the accuracy of CNN using random vector generation, skipgram and CBOW for 5 executions.

Performance comparison of proposed models with CNN is presented in Figs. 9, 10, 11, 12, 13 and 14. It is proved from the results that the proposed models produced more accurate results than CNN for 9 out of 10 datasets. The methods which generated highly accurate results are based on the combination of Synset and thesaurus. Major reason for the improvement in the results by the proposed models is that the synonyms are included which provides various combinations of words for test review bigrams. These combinations represent the same sentiments with the help of different words. Also, Synset based model outperforms...
all the other methods. Though the vocabulary of Syn-set is smaller when compared to Word2Vec, it has given significant improvement in the results as it contains a perfect set of synonyms for the words passed to it.
Discussion

Experimental results show that the proposed models perform better than its predecessors in 90% of the cases. The proposed models incorporate an improvement in the classification of product reviews based on synonyms. It retrieves semantically similar words from the thesaurus for sentiment-oriented words and from Synset and Word2Vec for both sentiment and non-sentiment words. As the sentiment words carry more weightage in finding sentiment of the review, we have also used Synset and Word2Vec only for sentiment words. The objective of this improvement is to produce more accurate results. From the results, it is observed that there is an improvement when using proposed models for extracting synonyms for the words present in the reviews. With all the measures presented in the paper, Synset-based methods (M2, M5, M7, M10 and M11) work well. When considering accuracy, M11 produces better results than M2, M5, M7 and M10. When F-measure is considered, performance of M11 and other synset-based methods are same. In all the other measures, M10 works well. The meaning of this is that more positive reviews are correctly classified using M10. But when

Fig. 3 Comparison of proposed models with NB using precision

Fig. 4 Comparison of proposed models with NB using recall
Fig. 5 Comparison of proposed models with NB using F-measure

Fig. 6 Comparison of proposed models with NB using completeness

Fig. 7 Comparison of proposed models with NB using false alarm rate
considering both positive and negative reviews, M11 works well.

The improvements in the results are due to the fact that synonyms for sentiment words in a thesaurus are good and Synset has better semantically similar words when considering both sentiment and non-sentiment words. Also, various reviewers use different words with similar meaning for representing their sentiment. For accommodating this case, bigrams are formed by using not only the words in test reviews but also using their synonyms.

Fig. 8 Accuracy of 5 runs or CNN for 10 datasets
Fig. 9  Comparison of proposed models with CNN using accuracy

Fig. 10  Comparison of proposed models with CNN using precision

Fig. 11  Comparison of proposed models with CNN using recall
Findings

At the outset, there are three methods in which synonyms of the words are extracted. They are sentiment thesaurus, Synset and Word2Vec. The conjecture from the results is that Synset is ideal for improved bigram model. When dividing the words into sentiment and non-sentiment words, a combination of thesaurus for sentiment words and Synset for non-sentiment words and only Synset for sentiment words are ideal choices. As the thesaurus contains synonyms only for sentiment words, thesaurus only method failed to work well. But it guarantees to generate a good classification model when combined with Synset.

When we analyze the performance of Word2Vec models, performance is better when it is combined with a thesaurus by applying sentiment words and non-sentiment words segregation. As the accuracy measure considers both positive and negative class reviews equally, it is guaranteed that the thesaurus and Synset combination is the best choice if the application treats both classes equally. Since the other measures give more weightage for positive class and Synset only model and the one which utilizes thesaurus, Synset and Word2Vec for sentiment words produced better results in those measures, it can be used in the applications where positive reviews are more important than negative reviews.
Product review classification assigns a category for the reviews based on the text content present in it. This task is necessary for the current era since it finds the attitude of users towards a particular product. In this paper, we proposed different models for classifying reviews to include semantically similar words for the words in the test bigrams using thesaurus, Synset and Word2Vec. The performance of the proposed models is compared with the traditional bigram model and the results are projected. From the results, it is proved that the proposed Synset with and without thesaurus is superior to the traditional bigram model and Word2Vec with and without thesaurus. We have also compared with other recent works on sentiment analysis using unigram and bigram with NB and CNN and observed that the proposed models are much better than those methods. In future, SentiWordnet score can also be combined with the bigram model to perform classification. Also synonym inclusion can be tried with n-gram model instead of the bigram model. PoS tagging can also be utilized in future to identify bigrams which are having same tags.

Conclusion

Product review classification assigns a category for the reviews based on the text content present in it. This task is necessary for the current era since it finds the attitude of users towards a particular product. In this paper, we proposed different models for classifying reviews to include semantically similar words for the words in the test bigrams using thesaurus, Synset and Word2Vec. The performance of the proposed models is compared with the traditional bigram model and the results are projected. From the results, it is proved that the proposed Synset with and without thesaurus is superior to the traditional bigram model and Word2Vec with and without thesaurus.

We have also compared with other recent works on sentiment analysis using unigram and bigram with NB and CNN and observed that the proposed models are much better than those methods. In future, SentiWordnet score can also be combined with the bigram model to perform classification. Also synonym inclusion can be tried with n-gram model instead of the bigram model. PoS tagging can also be utilized in future to identify bigrams which are having same tags.

**Funding**  Not applicable.

**Available of data and material**  Not applicable.

**Code availability**  Not applicable.

**Declarations**

**Conflict of interest** The authors declare that they have no conflict of interest.

**Ethical approval** This article does not contain any studies with human participants or animals performed by any of the authors.

**Informed consent**  Not applicable.

**References**

1. Pang B, Lee L, Vaithyanathan S. Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of Emnlp, p. 79–86, 2002.
2. Salvetti F, Relchenbach C, Lewis S. Opinion polarity identification of movie reviews. In: Computing attitude and affect in text: theory and applications. Chapter 23, p. 303–16.
3. Beineke P, Hastie T, Manning C, Vaithyanathan S. Exploring sentiment summarization. In: AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications, 2004.
4. Sahu T, Ahuja S. Sentiment analysis of movie reviews: a study on feature selection and classification algorithms. In: International Conference on Microelectronics, Computing and Communications (MicroCom), p. 1–6, 2016.
5. Bodapati I, Veeranjaneulu N, Shaik S. Sentiment analysis of movie reviews using LSTMs. Ingénierie des systèmes d’information. 2019;24(1):125–9.
6. Tripathy A, Agarwal A, Rath SK. Classification of sentiments using n-gram machine learning approach. Expert Syst Appl. 2016;57:117–26.
7. Vinodhini G, Chandrasekaran RM. A comparative performance evaluation of neural network based approach for sentiment classification of online reviews. J King Saud Univ Comput Inf Sci. 2016;28(1):2–12.
8. Bakliwal A, Patil A, Arora P, Varma V. Towards enhanced opinion classification using NLP techniques. In Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology (SAAIP 2011), 101–7, Chiang Mai, Thailand, 2011.
9. Acosta J, Lamaute N, Luo MX, Finkelstein E, Cotoranu A. Sentiment analysis of Twitter messages using Word2Vec. Student-Faculty Research Day, CCIS, Pace University, c8-1–c8-7, 2017.
10. Bansal B, Srivastava S. Sentiment classification of online consumer reviews using word vector representations. In: International Conference on Computational Intelligence and Data Science (ICCIDS 2018).

11. Mohammed B, Fatima SS. Using skip gram, n gram, and Part of Speech features for sentiment classification of Twitter messages. In: ICON, 2015.

12. Awachate P, Vivek B, Kshirsagar P. Improved Twitter sentiment analysis using N gram feature selection and combinations. Int J Adv Res Comput Commun Eng. 2016;5(9):154–7, ISO 3297:2007 Certified.

13. Hameed MA, Hussain AR, Sayeedunnissa SF. Sentiment analysis using Naive Bayes with Bigrams. Int J Adv Comput Sci Appl IJCSIA. 2014;4(2):84–7.

14. Thet TT, Na JC, Khoo CSG. Aspect based sentiment analysis of movie reviews on discussion boards. J Inf Sci. 2010;36(6):823–48.

15. Singh J, Singh G, Singh R. Optimization of sentiment analysis using machine learning classifiers. Hum Cent Comput Inf Sci. 2017;7:32.

16. Tsutsumi K, Shimada K, Endo T. Movie review classification on a multiple classifier. In: PACLIC, 2007.

17. Jang B, Kim I, Kim JW. Word2Vec convolutional neural network for classification of news articles and tweets. PLoS ONE. 2019;14(8):e0220976.

18. Sasikala P, Mary Immaculate Sheela L. Sentiment analysis of online product reviews using DLMNN and future prediction of online product using IANFIS. J Big Data. 2020;7:33.

19. Fauzi MA. Word2Vec model for sentiment analysis of product reviews in Indonesian language. Int J Electr Comput Eng (IJECE). 2019;9(1):525–30.

20. Poomagal S, Malar B, Inamul Hassan, Kishor R. A novel Tag_Score (T_S) model with improved K-means for clustering tweets. Sadhana Indian Acad Sci. 2020;45:1–13. (Article ID : 0125).

21. Zhang S. Sentiment classification of news text data using intelligent model. Front Psychol. 2021;12:1–9.

22. Fei R, Yao Q, Zhu Y, Xu Q, Hu B. Deep learning structure for cross-domain sentiment classification based on improved cross entropy and weight. Sci Program. 2020;2020. (Article ID : 3810261).

23. Souma W, Vodenska, I. and Aoyama, H. : Enhanced news sentiment analysis using sentiment methods. Journal of Computational Social Science, Vol. 2, 33-46, 2019.

24. Chandra, R. and Krishna, A. COVID-19 sentiment analysis via deep learning during the rise of novel cases. PLoS ONE. 2021;16(8):e0255615.

25. Miranda E, Aryuni M, Hariyanto R, Surya ES. Sentiment analysis using Sentiwordnet and machine learning approach (Indonesia general election opinion from the twitter content. In: International Conference on Information Management and Technology, 2019.

26. Liu B, Hu M, Cheng J. Opinion observer: analyzing and comparing opinions on the web. In: Proceedings of the 14th International World Wide Web Conference (WWW-2005), May, 10–14, 2005, Chiba, Japan.

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.