International Journal of Software Innovation
Volume 11 • Issue 1

SentiNeg:
Algorithm to Process Negations at Sentence Level in Sentiment Analysis

Sandhya R. Savanur, GM Institute of Technology, India*
R. Sumathi, Siddaganga Institute of Technology, India

ABSTRACT

Sentiment analysis is the process of identifying and categorizing opinions computationally to determine the attitude expressed in the spoken or written text as positive, negative, or neutral. Negation analysis is the task of analyzing the negative opinions by identifying the scope of negation within a sentence and applying linguistic or grammatical rules of the language. In this paper, the rules for identifying the scope of negation within a sentence and the rules applicable to different negation categories are defined. An algorithm by the name SentiNeg has been proposed for processing negations at the sentence level. SentiNeg algorithm filters non-opinionated sentences from the data to avoid unnecessary processing. For opiniated sentences, the algorithm applies different linguistic or grammatical rules of the language to identify negative opinions. SentiNeg algorithm takes opinionated sentences as input and provides a detailed aspect-based summary of negative opinions that are expressed on the entity under analysis.

KEYWORDS

Aspect-Based, Grammatical Rules, Linguistic Rules, Negation Handling, Opinionated Sentences, Sentence Level, Sentiment Analysis

1. INTRODUCTION

With the advancement in technology and rapid increase in social media platforms, the distance between businesses and customers is reducing. Customer feedback plays an important role in business. It gives the customer satisfaction level with respect to their product and services which helps in improvisation of the products or services.

The data available on the web is huge and it is present either in a structured or unstructured format. This data includes reviews, opinions, blogs, etc. Customer feedback is usually unstructured. To interpret the unstructured data and understand the polarity of the opinions expressed, it is important to leverage sentiment analysis.

DOI: 10.4018/IJSI.315741 *Corresponding Author

Sentiment refers to a human attitude, opinion and emotions that are qualitative objects. Sentiment analysis is the process of analyzing an individual's opinions, sentiments, attitudes and emotions towards entities such as products, services, organizations, individuals and events (Liu, 2012). Sentiment analysis interprets and classifies the emotions within the text using text analysis techniques. The task of sentiment analysis identifies positive, negative and neutral opinions, emotions and feelings from written information.

Sentiment analysis can be performed at three levels i.e., document level, sentence level and aspect level (Joshi & Itkat, 2014). Document-level sentiment analysis determines the overall opinion of the document. An assumption is made that each document expresses opinions on a single entity. Sentence level sentiment analysis determines opinions expressed in a sentence. Aspect level sentiment analysis performs a granular analysis and requires the use of different linguistic and grammatical rules. In sentiment analysis, it is important to identify positive, negative and neutral feelings expressed about the entity under analysis.

Negation analysis is one of the important subtasks of sentiment analysis. It is a complex task of identifying the negative opinions expressed within a context, which depends on many linguistic or grammatical rules of the language and negation scope identification. If not handled correctly, the presence of negation words within a sentence changes the polarity of the complete sentence. This, in turn, affects the results of sentiment analysis. Handling negation efficiently gives correct feedback about the entity and provides scope for improvement.

Sentiment analysis is considered as one of the text classification tasks in which the review text is represented by a bag-of-words (BOW) model. Machine learning algorithms such as Naïve Bayes or Support Vector Machine (SVM) are used for classification. But BOW model discards the word order and hence results in the loss of semantic information of the text. It results in the polarity shift problem, which is a phenomenon that reverses the polarity of the sentiment by linguistic structures called polarity shifters.

In lexicon-based sentiment analysis, negation handling techniques exploit grammatical relations among the words in a sentence and produce a dependency-based parse tree. The dependency-based parse tree is used to get the scope of negation and thus invert the polarity of words that are affected by negation. But these techniques ignore the consideration of features such as the influence of conjunctions and punctuation marks.

There are negation forms beyond the prototypical negations "not" and "never". These negation forms are called approximate negators. Approximate negators include words such as "rarely", "hardly", "seldom", "few", "little" etc. Approximate negators do not give the effect of absolute negators such as "not" and "never". Hence, handling approximate negators in identifying negation scope is challenging.

There are adjectives or adverbs, that describe another word or change its meaning in some way. Such words are called modifiers. Modifiers include negations, amplifiers and downtoners. Though negations are widely studied, amplifiers and downtoners need extensive attention.

In this paper, different categories of negation words that can exist within a sentence are explained. Rules applicable to different types of negation words and negation scope detection have been described. An algorithm called SentiNeg is proposed to identify and perform negation analysis on both simple and compound sentences. The algorithm provides a clear aspect-based summary of negative opinions expressed on the entity under analysis.

The remainder of the paper is organized as follows. Section 2 gives a brief overview of the related work in the area, Section 3 explains the concept of negation analysis, Section 4 gives negation analysis model, Section 5 explains the working of the proposed SentiNeg algorithm, Section 6 details about the results and analysis and finally Section 7 concludes the paper followed by future enhancements.

2. BACKGROUND

As per (Asmi & Ishaya, 2012), existing sentiment analysis approaches present negation identification and calculation methods using a developed framework for sentiment analysis that includes a description of negation rules. But the rules for propositional negation calculation have to be included. Sentiment analysis extracts information from positive and negative words in a text, from the context and the linguistic structure of the text. (Toboada, 2015) state that good knowledge about the language contributes in automatically determining a sentiment. Polarity shift refers to a linguistic phenomenon in which the polarity of the sentiment can be reversed by special linguistic structures called polarity shifters. (Xia et al., 2016) show that the polarity shift problem that occurs commonly in document-level sentiment analysis is addressed using a three-stage cascade model named Polarity Shift Detection, Elimination and Ensemble.

Machine learning approaches are found useful in negation analysis. (Cruz et al., 2016) describe a two-step machine-learning approach to automatically identify negation and speculation cues in the review domain in which the negation and speculation cues are identified and the full scope of these cues is determined. Improvisation is required in terms of scope detection results. Using machine learning techniques and methods for sentiment analysis poses many challenges. (Kharde & Sonawane, 2016) conducted a detailed survey and comparative study of existing machine learning and lexicon-based approaches for opinion mining with cross-domain and cross-lingual methods and evaluation metrics are conducted.

(Kiritchenko & Mohammed, 2016) show that negators, modals and degree adverbs affect the sentiment of the words that they modify. To aid this, a dataset of phrases is used which includes different negators, modals and degree adverbs and their combinations. (Fancellu et al., 2016) have tried neural networks' sequence-to-sequence models for detecting the scope of negation. Its performance on different categories of data, containing different types of negation has been detailed. The performance can be improved by adding language-independent structural information for exact scope matching. (Diamantini et al., 2016) have discussed a negation handling algorithm that automatically detects the scope of negation based on dependency-based parse trees. The functionality of the algorithm can be improved by adding features such as the influence of different conjunctions and considering the use of punctuation marks.

The problem of negation scope identification to identify the polarity of the sentence has been explained. (Farooq et al., 2017) have proposed a method for negation handling based on different linguistic features that determine the effect of different types of negation. Few exceptions to be taken care of include, negations that do not have any scope in the clause and the polarity of the clause getting inverted though there is no opinionated word in the clause. (Peng et al., 2018) have proposed an algorithm called NegBio to detect negative and uncertain findings in radiology reports. The algorithm uses patterns on universal dependencies to identify the scope of triggers that are indicative of negation or uncertainty.

If the traditional method of negation handling using the 'NEG_' tag for words is used, irrelevant words get negated. To handle this, (Amalia et al., 2018) have proposed two methods for negation handling namely representing negation and word negated to determine the scope of negation. But this method is domain-specific and restricted to the Indonesian language. (Hussein, 2018) has conducted and presented a survey on challenges faced in sentiment analysis relevant to their approaches and techniques. Regarding emotion detection, methods have been discussed to classify text into six different categories of emotions that are happiness, sadness, fear, anger, surprise and disgust. To eliminate the need for manual annotation of large datasets, (Gaind et al., 2018) have developed a method for the creation of a training set automatically. The discussed method is restricted to classifying six different emotions and does not classify a text as positive or negative.

(Palomino, 2018) has discussed that in modelling the automatic detection of negation in tweets, the word set plays an important role. Negation handling in dialogue acts prediction tasks on customer care conversations in Twitter is handled by creating a negation corpus and developing several heuristics for detecting negation scope in customer care conversations which is discussed by (Bhuiyan, 2018). (Ljajic & Morovac, 2018) have proved that the application of different rules of negation for processing tweets or short texts greatly improves the prediction of sentiment. The discussed rules can be improvised by expanding the dictionary of synonyms by using an existing morphological dictionary and analyzing the effect of intensifiers.

(Strohm & Kilnger, 2018) have discussed that it is very important to have a good understanding of modifiers such as amplifiers, downtoners and negations in the different contexts and their effect on document-level sentiment. (Fancellu et al., 2018) conducted a study at the document level which can be expanded to word and sub-word levels. Different neural models have been developed that learn from cross-lingual word embeddings or universal dependencies in English and test them on Chinese.

(Raheem, 2020) has discussed regarding using sentiment analysis in the healthcare domain. It highlights techniques, opportunities, challenges and future work by using sentiment analysis in the healthcare domain. (Zafra et al., 2020) have implemented negation detection for sentiment analysis and detailed error analysis in which a machine learning negation processing system is applied to the sentiment analysis task. Improvisations in this system include, correct identification of contiguous and non-contiguous cues, and developing a post-processing algorithm to cover the three types of scopes that can be found before the cue, after the cue, or before and after the cue. (Mohammed, 2021) summarizes the problems and applications associated with automatic sentiment analysis. The key challenges for automatic systems, algorithms, features, and datasets that are used in sentiment analysis are explained.

(Lopez et al., 2021) proposes a novel method to perform multimodal sentiment classification. The method includes individual text and image classification along with AutoML-generated model to perform final sentiment classification. (Tavchioski et al., 2022) explored three different models to detect depression from social media text. They include BERT family models, AutoML approaches and knowledge-based representations based on knowledge graph concepts. (Siriborvornratanakul, 2022) conducted a case study to investigate the gap between AutoML frameworks and practical machine learning development. (Gomez-Espinos et al., 2022) conducted experiments to show the effectiveness of using high level features from text learn from BERT models.

Existing negation handling techniques provide methods for analyzing negations on data available on the web. They provide methods for corpus creation that will be used to identify negations. But there are different types of negations present in the language and it is necessary to understand their position, meaning and context. The requirement of detailed analysis of negations as per the linguistic or grammatical rules of the language still exists. In this paper, the negation words are grouped into different categories. Negation rules that consider the grammatical and linguistic rules of the language are proposed to perform negation analysis. A negation handling algorithm SentiNeg is proposed for sentence-level sentiment analysis, which performs negation handling on the opinionated sentences. SentiNeg algorithm considers different linguistic or grammatical rules of the language that apply to the types of negation words used in the sentence. SentiNeg also gives a clear summary of negative opinions expressed on the entity under analysis.

3. NEGATION ANALYSIS

Negation is a grammatical construction that contradicts all or part of the meaning of a sentence. Negation words are the most important class of sentiment shifters (Liu, 2012). Negation analysis is an automatic way of determining the scope of negation and inverting the polarities of opinionated words that are affected by the negation. The portion of the sentence that is affected by the negation is called the scope of negation.

Section 3.1 explains the types of negations that are encountered at the sentence level along with appropriate examples. Types of negations are grouped into different categories to apply negation rules. Different categories of negations are described in Section 3.2. The rules used for negation analysis are explained in Section 3.3.

3.1 Types of Negations

Negations that are encountered at the sentence level are classified into six types. Table 1 gives the details about the types of negation with appropriate examples.

3.2 Negation Categories

Categorization of negations makes it convenient to apply relevant negation rules. Figure 1 shows the types of negations and their mapping to different negation categories.

Negation categories and their definitions are explained in Table 2.

3.3 Rules for Negation Analysis

In-depth knowledge of the language is necessary to perform negation analysis. It is important to understand the usage and context of negation words used in a sentence. The usage and context of the different negation words were studied and rules are framed to aid in performing negation analysis. In this section, the rules that can be applied to negation categories explained in section 3.2 are discussed.

3.3.1 Rules for Analyzing Punctuations

Punctuation marks play a very important role in defining the scope of negation.

Table 3 explains the negation rules associated with different punctuation marks.

3.3.2 Rules for Analyzing Downtoners

Table 4 explains the rules for analyzing downtoners.

Table 1. Types of Negation with examples

Type of Negation	Definition/Usage	Example
Negation forms of the verb	By adding the negative particle 'not' before a verb, the meaning of the verb is negated. In certain contexts, 'not' is replaced by 'never'.	1. I did not like the product.
	Principal/Action verbs: These verbs are the main verbs that express the action of the subject. The verbs that express action are called action verbs.	run, walk, eat, dance, etc
	Auxiliary verbs: These verbs are the supporting verbs that are used together with the main verb to determine the verb's tense, moods or to form a question or negative sentence.	do, be, have
	Modal Auxiliary verbs: These verbs are auxiliary verbs that never change their form.	can, could, may, might, shall, etc
Negation with noun or pronoun	The negative particle 'no' can be attached to the noun group, either to the subject or direct object of a sentence to obtain negative meaning.	1. No cellphones of that company have good battery backup.
	'No' can be combined with words like –one, -thing, -where, etc to make indefinite pronouns like noone, nothing, nowhere etc.	1. There is nothing good with that product.
Downtoners	Downtoners are the words that reduce the polarity of other words or phrases instead of completely inverting the polarities. Hardly, rarely, little, less, etc are a few common downtoners.	1. This cellphone hardly works.
Negation using adverb phrase	A negative meaning can be added to a sentence by including an adverb phrase with a negative meaning. The most common group of adverbial phrases is formed using the word 'without' or a preposition followed by 'no'.	1. This cellphone hangs for no reason at all.
Negation using 'Neither-Nor'	'Neither' and 'nor' are used to link two negative pronouncements. They can be attached to verbs, nouns and even to prepositional phrases.	1. This fabric is neither soft nor of good quality.
Negation using negative adjectives	An affirmative statement can be made into a negative statement by adding a negative prefix (im, un, in, dis) or suffix(less) to an appropriate adjective.	1. This handset is useless.

Figure 1. Types of negations and their mapping to negation categories

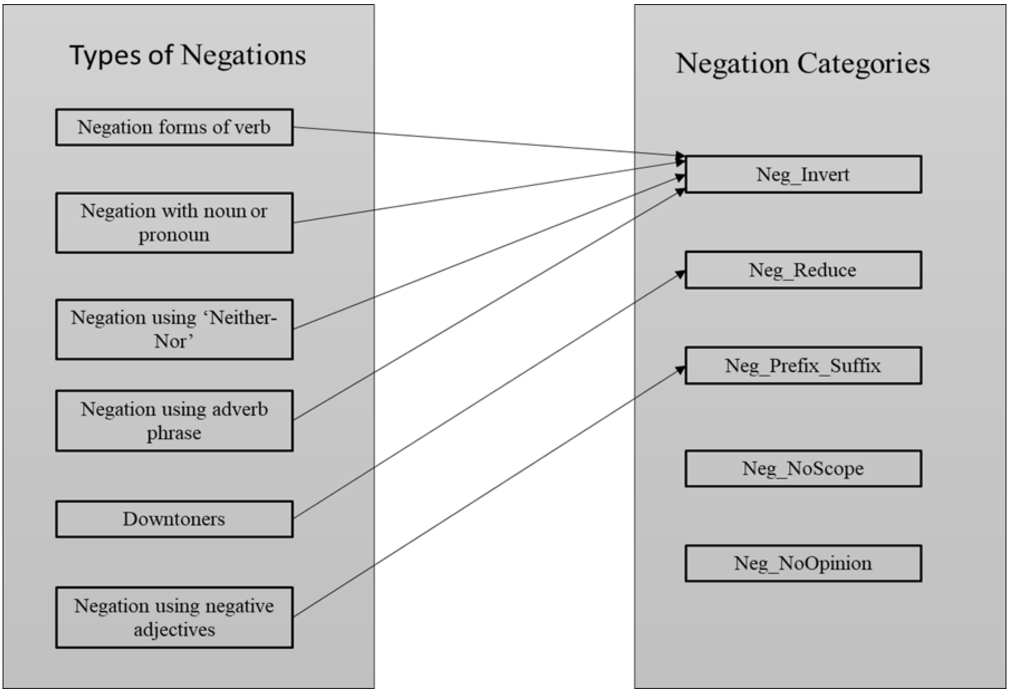


Table 2. Negation categories and their definition

Negation Category	Definition
Neg_Invert	Types of negations that invert the polarity of other words fall within the group Neg_Invert. It includes, 1. Negation forms of a verb 2. Negation with noun and pronoun 3. Negation using adverb phrase 4. Negations with neither – nor
Neg_Reduce	Negation words that reduce the polarity of other words or phrases instead of completely inverting the polarities come under Neg_Reduce. All the downtoners fall under the Neg_Reduce category.
Neg_Prefix_Suffix	Neg_Prefix_Suffix group includes all the negation words formed by adding prefix or suffix to the appropriate words. Negation words with negative adjectives fall under the Neg_Prefix_Suffix category.
Neg_NoScope	All those words that do not change their polarities even when they appear after a negation word are grouped under the Neg_NoScope group. There is no scope for negation though these words follow a negation word. Example: 'not just', 'no wonder' etc.
Neg_NoOpinion	All those nouns that appear after a negation word but do not express any opinion fall under the Neg_ NoOpinion group. Example: 1. There is no Bluetooth in this cellphone.

3.3.3 Rules for Analyzing Words with Negative Prefix Or Suffix

Table 5 explains the rules for analyzing words with negative prefix or suffix.

3.3.4 Rules for Analyzing Negations Under Neg_NoScope Category

Table 6 explains the rules for analyzing negations under the Neg_NoScope category.

Table 3. Punctuation marks, their usage and rules associated with them

Category	Punctuation mark	Usage	Rule	Example
Neg_Invert	Period or Fullstop (.)	End the sentence	If a period is encountered, a check has to be performed to find out if it is a simple or compound sentence. If it is a compound sentence, then the sentence has to be split on the conjunction being used, as per the rules discussed in the FBSACS algorithm (Savanur & Sumathi, 2017).	1. I did not like the product. 2. The fabric is rough and the color is dull.
	Question mark (?)	End an interrogative sentence or a tag question	Sentences with tag questions might contain opinions. The opinions expressed in sentences with tag questions need to be extracted.	The cellphone is not good. Isn't it?
	Exclamation mark (!)	Used at the end of the statement. It adds a strong feeling or urgency to the sentence. It expresses a warning, an order or strong feelings like surprise, fear, happiness, etc.	The words expressing the opinion about the entity need to be extracted from the sentence.	The product is damaged!
	Colon (:)	Introduce a word, words, a phrase, a list or a quotation.	When a colon is used, the opinion might be present on both sides of the colon. So, it is necessary to extract opinion-expressing words from both sides of the colon.	I have only one thing to say about this product: Worst.
	Semicolon (;)	Connect two related or similar sentences or connect items in a list, if there are already commas in a sentence.	When a semicolon is used, the opinion might be present on both sides of the semicolon. So, it is necessary to extract opinion expressing words from both sides of the semicolon.	The mobile is not working; I need to buy a new one.
Neg_Invert	Apostrophe (')	Indicates the ownership by a noun or represents missing letters in a contraction.	If an apostrophe is encountered, its preceding word is the object on which the opinion is expressed. The contraction is when two words are combined to make one shorter word like can't, haven't, etc	The mobile's battery isn't working.
	Comma (,)	1. Separate words or phrases when listing items in a sentence. 2. Separate multiple adjectives describing the same object. 3. Separate two contrasting phrases	If a comma is encountered in a sentence, the noun, adjectives and verbs need to be extracted to analyze the opinion.	The cellphone is handy, sleek and light weighted.

Table 4. Definition and rules for analyzing downtoners

Category	Definition	Rule
Downtoners	Downtoners reduce the polarity of other words or phrases instead of completely inverting the polarities. The downtoners are usually adverbs. They either precede or follow the adjective or action verb that it modifies in the sentence or it affects the nearby adjective or action verb in the sentence.	During negation analysis, if a downtoner is encountered, the effect of the accompanying adjective or action verb is reduced by 50%.

3.3.5 Rules for Analyzing Negations Under Neg_NoOpinion Category

Table 7 explains the rules for analyzing negations under the Neg_NoOpinion category.

3.3.6 Rules for Analysis of Simple and Compound Sentences

Table 8 explains the rules for analyzing simple and compound sentences.

Table 5. Definition and rules for analyzing words with negative prefix or suffix

Category	Definition	Rule
Negative Prefix	A negative prefix is a group of letters added to the front of the root word to change its meaning. It carries a negative meaning 'not' or 'opposite of'. Example: un-, im-, in-, il-, ir-, dis-, de-, mis-, non-	For negation analysis of negative prefixes and suffixes, all the root words specific to the domain under analysis, are collected along with their polarity and stored in a root word dictionary (RW-Dict). When such a negation word is encountered, the word is split to obtain the root word. The polarity of the root word is obtained from the RW-Dict and the polarity is inverted.
Negative Suffix	A negative suffix is a group of letters added to the end of the root word to change its meaning. It carries a negative meaning 'absence of'. Example: -less	

Table 6. Definition and rules for analyzing negations under Neg_NoScope category

Category	Definition	Rule
Neg_ NoScope	All those words that do not change their polarities even when they appear after a negation word fall under this category. There is no scope for negation though these words are prefixed with negation words. Example: 'not just', 'no wonder' etc	For negation analysis of such cases, we eliminate those words that appear after a negation word if they are not adjectives, action verb or a noun.

Table 7. Definition and rules for analyzing negations under Neg_NoOpinion category

Category	Definition	Rule
Neg_ NoOpinion	All those nouns that appear after a negation word but do not express any opinion fall under this category. Example: 1. There is no Bluetooth in this cellphone.	For negation analysis of such cases, a domain-specific noun dictionary (N_Dict) is maintained. When such a noun, following a negation word, is encountered in a sentence, its presence is first checked in N_Dict. If it is present, then such a sentence is assumed to have a negative opinion.

Table 8. Definition and rules for analyzing simple and compound sentences

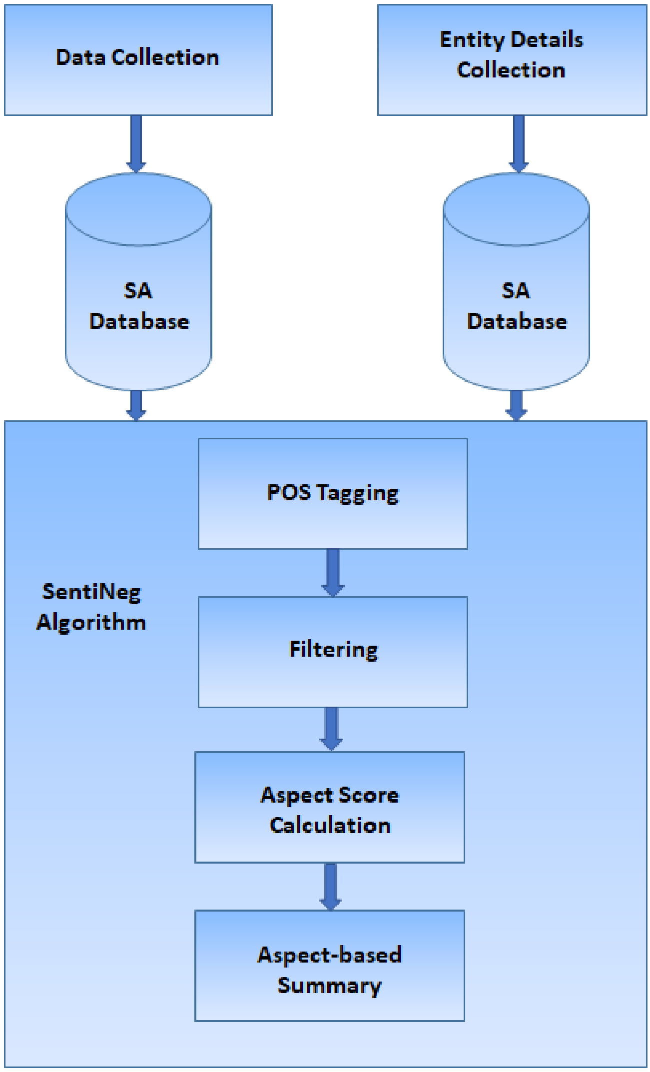
Category	Sentence Type	Definition	Rule
Neg_ Invert	Simple Sentence	A simple sentence consists of one clause with a single subject and predicate. Example: 1. I did not like the texture of this fabric.	Apply the rules defined in the above sections to perform negation analysis.
	Compound Sentence	A compound sentence consists of multiple clauses joined together with conjunctions. Example: The fabric is rough and the color is dull.	As the first step, the conjunctions used in the sentence have to be identified. Then the clauses that the conjunction connects have to be independently analyzed as per the grammatical rules applicable to them (Savanur & Sumathi, 2017). We refer FBSACS algorithm discussed in (Savanur & Sumathi, 2017) to perform conjunction analysis. After conjunction analysis, apply the rules defined in the above sections to perform negation analysis.

4. NEGATION ANALYSIS MODEL

The block diagram of the negation analysis model is shown in Figure 2.

The negation analysis model has four components and the functionality of each component of the model is given below.

Figure 2. Negation analysis model



4.1 Data Collection

The data required for sentiment analysis should be specialized and be available in large quantities. Finding relevant datasets is a challenging task. The data sources include organization databases, reviews or feedback from websites, training datasets, blogs and flat files. The candidate data for negation analysis is collected from the concerned website using a web scraper or from a domain database and other sources using ETL tools. The collected data is stored in the SA database for further processing.

4.2 Entity Details Collection

The entity details collection stage is a preparation stage in which all the details required by the proposed SentiNeg algorithm, are collected and stored in a single database. The aspects and all the details related to the entity, together form the aspect dictionary. Aspect dictionary is a repository for all the information required by the proposed SentiNeg algorithm to perform negation analysis.

Table 9 lists the tables accessed by the SentiNeg algorithm and their definition.

Table 9. Tables accessed by SentiNeg algorithm and their definition

Table Name	Definition
ASPECT _TAB	The features, parts or aspects of the entity under analysis, on which the negation analysis is to be performed, are collected and stored in the ASPECT_TAB table.
POL_TAB	The positive and negative words describing the aspects of the entity are collected and stored in the POL_TAB table. The synonyms for the positive and negative words are extracted using WordNet and are stored in the table POL_TAB.
NEG_TAB	The negation words/negative adverbs that are specific to the domain are collected and stored in the NEG_TAB table.
RW_DICT	The domain-specific root words used for analyzing negative prefixes and suffixes are stored in the RW_DICT table along with their polarities.
DOWNTONER	The downtoner words that are relevant to the domain are stored in the DOWNTONER table.
N_DICT	Nouns that are mandatory in the domain are stored in the N_DICT table.
CURSCORE	A table CURSCORE is created to store the negative score calculated for each aspect.

4.3 POS Tagging

In this stage, the proposed SentiNeg algorithm performs Part of Speech tagging for each word within the sentence of candidate data. As a result, part of speech is assigned to every word in the sentence.

4.4 Filtering

After Part of Speech tagging is performed, the proposed SentiNeg algorithm checks each sentence for the presence of an adjective or action verb. If an adjective or action verb is absent in the sentence, then the sentence is considered to be a non-opinionated sentence. Such non-opinionated sentences are ignored and will not be processed further. This filtering of non-opinionated sentences avoids the unnecessary processing of sentences that are not the candidates for sentiment analysis. The output of the Filtering stage is the data with only opinionated sentences.

4.5 Aspect Score Calculation

The data with only opinionated sentences is provided as input to the Aspect Score Calculation stage.

Punctuations play a very important role in identifying the scope of negation. The proposed SentiNeg algorithm scans each opinionated sentence to find the presence of punctuation. As per the rules governing the usage of particular punctuation explained in section 3.3.1, the part of the sentence which is to be processed further is extracted. If there are no punctuations, the entire sentence is extracted.

The sentence is then checked if it is a simple sentence or a compound sentence. If it is a compound sentence, then it is split on the conjunction to form multiple clauses. In each clause, the noun, adjective, action verb and adverbs are collected.

The next set of validations include the presence of downtoners, negative prefix/suffix accompanying adjective/action verb and negation words/negative adverbs. The proposed SentiNeg algorithm considers the grammatical rules to perform these validations and calculates the negative score for each aspect-describing word (adjective, action verb, noun).

4.6 Aspect Based Summary

The output of the Aspect Score Calculation stage is the negative score for each adjective/action verb/noun that describes the aspect of the entity. The proposed SentiNeg algorithm consolidates these scores to output the negative feedback score for each component and aspect of the entity. This gives a detailed understanding of the performance and status of different aspects of the entity.

5. SENTINEG ALGORITHM

The proposed SentiNeg algorithm calculates the negative score of each aspect of the entity. The SentiNeg algorithm and its working is explained in this section.

5.1 Variables Used in SentiNeg Algorithm

Table 10 lists the variable names used in the SentiNeg algorithm and their definitions.

5.2 Working of SentiNeg Algorithm

5.2.1 POS Tagging and Filtering

SentiNeg algorithm performs Part of Speech tagging for each word of every sentence extracted from the SA database. If the sentence does not contain an adjective or action verb, the sentence is considered non-opinionated and is ignored from further processing. The process of filtering out non-opinionated sentences avoids the unnecessary processing of sentences that are not useful for sentiment analysis.

Table 11 shows the algorithm for POS tagging and filtering.

5.2.2 Punctuation Validation

Each opinionated sentence is checked for the presence of punctuations and part of the sentence is extracted as per rules explained in section 3.3.1. Table 12 shows the algorithm for punctuation validation.

Table 10. Variable names and their definitions

Variable Name		Definition
S	Variable to store the	sentence selected from the database.
S_n		extracted sentence
a_score		score of negative adjectives
v_score		score of negative verbs
n_score		score of negative nouns
N		noun present in the selected sentence
A_W		adjective present in the selected sentence
V_W		action verb present in the selected sentence
NG_W		negative adverb/negation word present in the selected sentence
W		word with negative prefix or suffix
P		polarity of the word with negative prefix or suffix
N		noun following a negation word/negative adverb
current_score		Variable to store the current score of the aspect from the CURSCORE table.
N	Variable to copy	noun parameter in findscore function
v		action verb parameter in findscore function
A		adjective parameter in findscore function
Nw		negation word/negative adverb parameter in findscore function

Table 11. Algorithm to perform POS tagging and filtering

Input:	SA database consisting of sentences
Output:	Entity aspect and negative feedback score
	Begin
Step 1:	//Assigning Part of Speech to each word in a sentence and filtering non-opinionated sentences
	For (each *S* ← *select* sentence *from* SA database) *do* for all sentences
	Begin
	S_n ← “”; // Variable to store the extracted sentence
	a_score = 0; // Variable to store adjective score
	v_score = 0; // Variable to store verb score
	n_score = 0; // Variable to store noun score
	Assign Part of Speech to each word in the sentence
	If (S contains an adjective or action verb) *then*
	go to Step 2
	Else
	Discard S; // no adjective or action verb, discard the sentence
	End if
	End

5.2.3 Sentence Type Validation

The sentence can be simple or a compound sentence. The aspect that is being discussed in the sentence is present in the form of a noun. The noun is thus extracted from the sentence. The adjectives and action verbs in the sentence describe the aspect. Hence, the adjectives and action verbs are extracted. Negative adverbs are the negation words that are present in the sentence that are to be extracted. If the sentence is a compound sentence, the conjunctions present in the sentence are processed as per the FBSACS algorithm (Savanur & Sumathi, 2017) and validations on punctuations are repeated. Thus, the noun, adjectives, action verbs and the negation words /negative adverbs are extracted from the sentence. Table 13 shows the algorithm for sentence type validation.

5.2.4 Downtoners Validation

The sentence is checked for the presence of downtoners. If a downtoner accompanies an adjective/ action verb in the sentence, the effect of the adjective/ action verb is reduced by 50%. So, the respective score of the adjective/ action verb is set to 0.5 instead of 1. Table 14 show the algorithm for downtoners validation.

5.2.5 Validation of Words with Negative Prefix or Suffix

The sentence can have a negative prefix or suffix accompanying an adjective/action verb. In that case, the adjective/ action verb is extracted ignoring the prefix/suffix. The polarity of the adjective/ action verb is checked in the RW_DICT table. If the polarity is positive, the respective score of adjective/

Table 12. Algorithm for punctuation validation

Step 2:	// Check for punctuation in the selected sentence and copy the sentence
	// sentence ends with a period (.) and contains no other punctuation mark
	If ((S ends with ".") *and* (no other punctuation mark)) *then*
	$S_n \leftarrow$ S; // copy the sentence till the period
	//sentence contains a period (.) and ends with a question mark (?)
	Else If ((check for "." in S) *and* (S ends with "?")) *then*
	$S_n \leftarrow$ S; // copy the sentence till the period
	// sentence contains a period (.) and ends with an exclamation mark (!)
	Else If ((check for "." in S) *and* (S ends with for "!")) *then*
	$S_n \leftarrow$ S; // copy the sentence till the period
	//sentence contains an exclamation mark (!) and ends with a period (.)
	Else If ((check for "!" in S) *and* (S ends with ".")) *then*
	$S_n \leftarrow$ S; // copy the sentence between exclamation mark and period
	//sentence contains a colon (:) and ends with a period (.)
	Else If ((check for ":" in S) *and* (S ends with ".")) *then*
	$S_n \leftarrow$ S; // copy the sentence till the period
	//sentence contains a semicolon (;) and ends with a period (.)
	Else If ((check for ";" in S) *and* (S ends with ".")) *then*
	$S_n \leftarrow$ S; // copy the sentence till the period
	End if

action verb is set to 1 as there was a negative prefix/suffix attached to it. If the polarity is negative, the respective score of the adjective/action verb is set to 0. Table 15 show the algorithm for validation of words with negative prefix or suffix.

5.2.6 Validation of Negation Words/Negative Adverb Preceding a Noun

There could be a negation word/negative adverb preceding a noun in the sentence. The noun could be an aspect that has to be mandatorily present in the entity. Such nouns are stored in the N_DICT table. If the noun is present in the N_DICT table, the respective score for the noun is set as 1 as it is following a negative adverb. If the noun is not present in the N_DICT table, the respective score for the noun is set as 0. Table 16 shows the algorithm for validation of negation words/negative adverb preceding a noun.

5.2.7 Negative Feedback Score Calculation

Once the noun, adjective, action verb, negation word/negative adverb are extracted and respective scores calculated, the SentiNeg algorithm calls findscore function to calculate the negative feedback score of each aspect. Table 17 shows the algorithm for negative feedback score calculation.

Table 13. Algorithm for sentence type validation

Step 3:	// Check if S_n is a simple or compound sentence and extract noun, adjectives, action verbs and adverbs/negation words depending on the punctuations used
	N ⟵ “” //Variable to store the noun present in the sentence
	A_W ⟵ “” //Variable to store the adjective present in the sentence
	V_W ⟵ “” //Variable to store action verb present in the sentence
	NG_W ⟵ “” //Variable to store negative adverb/negation word present in the sentence
	// Check if S_n is a simple sentence
	If (S_n is a simple sentence) *then*
	// S_n does not contain an apostrophe (‘) and comma (,)
	If ((check for “’” not in S_n) *or* (check for “,” not in S_n)) *then*
	N ⟵ Noun in S_n //Copy noun to N
	A_W ⟵ Adjective in S_n //Copy adjective to A_W
	V_W ⟵ Action verb in S_n //Copy action verb to V_W
	NG_W ⟵ Negative adverb/negation word in S_n //Copy negative adverb/negation word to NG_W
	// S_n contains an apostrophe (‘) but not comma (,)
	Else If ((check for “’” in S_n) *and* (check for “,” not in S_n)) *then*
	N ⟵ Noun following the ‘s in S_n //Copy noun to N
	A_W ⟵ Adjective in S_n //Copy adjective to A_W
	V_W ⟵ Action verb in S_n //Copy action verb to V_W
	NG_W ⟵ Negative adverb/negation word in S_n //Copy negative adverb/negation word to NG_W
	//S_n contains a comma (,) but not an apostrophe (‘)
	Else If ((check for “’” not in S_n) *and* (check for “,” in S_n)) *then*
	N ⟵ Noun in S_n //Copy noun to N
	A_W ⟵ Adjectives in S_n //Copy adjectives to A_W
	V_W ⟵ Action verbs in S_n //Copy action verbs to V_W
	NG_W ⟵ Negative adverbs/negation words in S_n //Copy negative adverbs/negation words to NG_W
	// S_n contains both an apostrophe (‘) and a comma (,)
	Else If ((check for “’” in S_n) *and* (check for “,” in S_n)) *then*
	N ⟵ Noun following the ‘s in S_n //Copy noun to N
	A_W ⟵ Adjectives in S_n //Copy adjectives to A_W
	V_W ⟵ Action verbs in S_n //Copy action verbs to V_W
	NG_W ⟵ Negative adverbs/negation words in S_n //Copy negative adverbs/negation words to NG_W
	End if

continued on following page

Table 13. Continued

	// Check if S_n is a compound sentence
	Else If (S_n is a compound sentence) *then*
	Process conjunctions in S_n using FBSACS algorithm (Savanur & Sumathi, 2017)
	Goto Step 2 //analyze punctuations
	N ⟵ Noun in S_n //Copy noun to N
	A_W ⟵ Adjectives in S_n //Copy adjectives to A_W
	V_W ⟵ Action verbs in S_n //Copy action verbs to V_W
	NG_W ⟵ Negative adverbs/negation words in S_n //Copy negative adverbs/negation words to NG_W
	End if

Table 14. Algorithm for downtoners validation

Step 4:	//Check for downtoner accompanying any adjective or action verb in the sentence
	If (S_n contains a downtoner with an adjective) and (downtoner *in* DOWNTONER table) *then*
	a_score = 0.5; //Set adjective score to 0.5
	Else If (S_n contains a downtoner with an action verb) and (downtoner *in* DOWNTONER table) *then*
	v_score = 0.5; //Set verb score to 0.5
	End if

5.2.8 Display Negative Feedback Score

The SentiNeg algorithm then calls showscore function to print the negative feedback score of each aspect of the entity under analysis. Table 18 shows the algorithm to display negative feedback score.

6. RESULTS AND DISCUSSION

6.1 Working of SentiNeg Algorithm

The working of the proposed SentiNeg algorithm is explained in this section with an example review statement about the display aspect of a laptop.

Example: “The display is unclear and not working.”

6.1.1 POS Tagging

The part of speech is assigned to each word in the sentence and the output is shown in Figure 3.

6.1.2 Filtering

The sentence containing an adjective or an action verb is considered for further processing. Otherwise, the current sentence is ignored and the next sentence is selected from the SA database.

As the example contains both adjective and action verb, it is processed further.

6.1.3 Aspect Score Calculation

To calculate the negative score of the display aspect, the below steps are followed.

Table 15. Algorithm for validation of words with negative prefix or suffix

Step 5:	//Check if the sentence has a word with a negative prefix or suffix accompanying any adjective or action verb in the sentence
	If (S_n contains word W with the negative prefix 'n') *or* (S_n contains word W with the negative suffix 'n') *then*
	W ⟵ W – n; //Remove negative prefix/suffix from 'n' from word W
	P ⟵ Polarity of W; //Get polarity of W from RW_DICT table
	// Check for the positive polarity of W and set adjective score or action verb score to 1
	If (P > 0) *then* //check if polarity of W is positive
	If (W is an adjective) *then*
	a_score = 1; //Set adjective score to 1 if W is an adjective
	Else
	v_score = 1; //Set verb score to 1 if W is a verb
	End if
	// Check for the negative polarity of W and set adjective score or action verb score to 0
	Else If (P < 0) *then* //check if polarity of W is negative
	If (W is an adjective) *then*
	a_score = 0; //Set adjective score to 0 if W is an adjective
	Else
	v_score = 0; //Set verb score to 0 if W is a verb
	End if
	End if
	End if

Table 16. Algorithm for validation of negation words/negative adverbs preceding a noun

Step 6:	//Check whether the sentence has a noun following a negation word/negative adverb
	If (S_n contains noun "N" following a negation word/negative adverb 'n') *then*
	//Check if noun 'N' is present in the N_DICT table and 'n' is present in the NEG_TAB table
	If (N in N_DICT table) and (n in NEG_TAB table) *then*
	n_score = 1; //Set noun score as 1
	Else
	n_score = 0; //Set noun score as 0
	End if
	End if

Initialize adjective, verb and noun score to 0 as shown in Table 19.

1. The sentence is first checked for the presence of punctuations. As the sentence contains only period(.), the part of the sentence preceding the period is considered for processing (shown in step 2 of the SentiNeg algorithm).

Table 17. Algorithm for negative feedback score calculation

Input:	noun, n_score, adjective, a_score, action verb, v_score and negation word/negative adverb
Output:	Entity aspect and negative feedback score
Step 1:	//To calculate the score of each aspect of the entity under analysis
	n $\leftarrow$ noun //Copy noun parameter to variable n
	v $\leftarrow$ action verb //Copy action verb parameter to variable v
	a $\leftarrow$ adjective //Copy adjective parameter to variable a
	nw $\leftarrow$ negation word/negative adverb //Copy negation word/negative adverb parameter to nw
	current_score = 0; //Variable to store the current score
	//Check whether noun score is already assigned
	If (n_score $\neq$ 0) *then*
	//Get the noun score of the aspect from the CURSCORE table
	current_score $\leftarrow$ value of the score of noun 'n' from the CURSCORE table
	//Add current score and noun score
	n_score $=$ n_score $+$ currect_score
	End if
	//Check whether the adjective score is already assigned
	If (a_score $\neq$ 0) *then*
	//Get the noun score of the aspect from the CURSCORE table
	current_score $\leftarrow$ value of the score of noun 'n' from the CURSCORE table
	//Add current score and adjective score
	a_score $=$ a_score $+$ currect_score
	// Check whether the adjective score is not assigned
	Else if
	//Check if the adjective has positive polarity in the POL_TAB table
	If (a is positive) *then*
	//Check if negation word/negative adverb is present in NEG_TAB table
	If (nw $\neq$ "") *and* (nw in NEG_TAB) *then*
	a_score = a_score + 1 //Increment adjective score
	Else
	a_score = a_score - 1 //Decrement adjective score
	End if
	//Check if the adjective has negative polarity in the POL_TAB table
	Else If (a is negative) *then*
	//Check if negation word/negative adverb is present in NEG_TAB table

continued on following page

Table 17. Continued

	If (nw ≠ “”) *and* (nw in NEG_TAB) *then*
	a_score = a_score - 1 //Decrement adjective score
	Else
	a_score = a_score + 1 //Increment adjective score
	End if
	End if
	End if
	//Check whether verb score is already assigned
	If (v_score ≠ 0) *then*
	//Get the noun score of the aspect from the CURSCORE table
	current_score ← value of the score of noun ‘n’ from the CURSCORE table
	//Add current score and verb score
	v_score = v_score + currect_score
	// Check whether verb score is not assigned
	Else if
	//Check if the verb has positive polarity in the POL_TAB table
	If (v is positive) *then*
	//Check if negation word/negative adverb is present in NEG_TAB table
	If (nw ≠ “”) *and* (nw in NEG_TAB) *then*
	v_score = v_score + 1 //Increment verb score
	Else
	v_score = v_score - 1 //Decrement verb score
	End if
	//Check if the adjective has negative polarity in the POL_TAB table
	Else If (v is negative) *then*
	//Check if negation word/negative adverb is present in NEG_TAB table
	If (nw ≠ “”) *and* (nw in NEG_TAB) *then*
	v_score = v_score - 1 //Decrement verb score
	Else
	v_score = v_score + 1 //Increment verb score
	End if
	End if
	End if
Step 2:	//Calculate and store the score value in the CURSCORE table
	score = 0; //Variable to store the final score
	//Add noun score, adjective score and verb score to get the final negative score of the aspect

continued on following page

Table 17. Continued

	score = n_score + a_score + v_score
	Update the value of the score in the CURSCORE table for the noun 'n'.
	Call showscore function

Table 18. Algorithm to display negative feedback score

Output:	Print Entity aspect and negative feedback score
Step 1:	//Print the score of each aspect of the entity under analysis
	For (each aspect ← *select* aspect *from* ASPECT_TAB table) *do* for all aspects
	Begin
	Print aspect, score *from* CURSCORE table
	End

Figure 3. POS Tagging to each word in the sentence

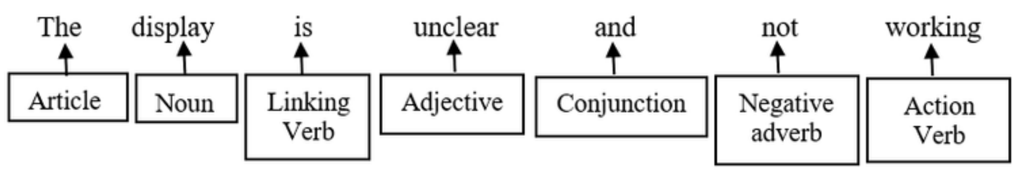


Table 19. Initialization of adjective, verb and noun score to 0

a_score = 0	//adjective score
v_score = 0	//action verb score
n_score = 0	//noun score

Output: The display is unclear and not working

2. A check is performed to identify if the sentence is a simple sentence or a compound sentence. The example sentence contains conjunction 'and' and hence identified as a compound sentence (shown in step 3 of SentiNeg algorithm). Each word in the sentence is assigned part of speech in the section 6.1.1. Extract nouns, adjectives, action verbs and negation words/negative adverbs from the sentence to the respective variables. Output of this step is shown in Table 20.
Output:
3. The sentence is scanned to check for the presence of downtoners. As there are no downtoners present in the example, this step is skipped (shown in step 4 of the SentiNeg algorithm).
4. This step checks for the presence of an adjective or action verb with a negative prefix or suffix. The example sentence contains an adjective 'unclear' with the negative prefix 'un'. The adjective

Table 20. Value of noun, adjective, action verb and negation word

Variable	Value
N	display
A_w	unclear
V_w	working
NG_w	not

is split on the prefix and the polarity of the adjective is fetched from the RW_DICT table. If the polarity is positive, the adjective score is set to 1 else it is set to 0 (shown in step 5 of SentiNeg algorithm).
Output of this step is shown in Table 21.
Output:

5. A check is performed to see if the sentence has a noun following a negation word/negative adverb. As the example sentence does not satisfy this scenario, this step is skipped (shown in step 5 of the SentiNeg algorithm).
6. Call findscore function with the noun, n_score, adjective, a_score, action verb, v_score and negation word/negative adverb to calculate negative feedback score for each noun.
Output: findscore (display, 0, unclear, 1, working, 0, not)
7. In this step, as the noun score for the display aspect, n_score is 0, the statements under the condition n_score $\neq$ 0 will not be executed. Since the adjective score for the adjective 'unclear', a_score is not equal to 0, the current negative score for the aspect 'display' is fetched from the CURSCORE table and added to the a_score.
Output:

Assuming current negative score of the aspect 'display' is 10 in CURSCORE table,

a_score = 1+10
a_score = 11

8. The action verb score v_score is 0 for the example sentence. The polarity of the action verb is fetched from the POL_TAB table. A check is performed to see if there is a negation word/negative adverb is present with the action verb and it is present in the NEG_TAB table. In the example, the action verb 'working' is preceded by a negative adverb 'not'.
Output:

Assuming the polarity of action verb 'working' is positive in POL_TAB table and 'not' is present in NEG_TAB table,

Table 21. Value of root word, polarity and adjective score after step 4

Variable	Value	Explanation
W	'clear'	W $\leftarrow$ unclear – un
P	1	Polarity of 'clear' from RW_DICT table. Assuming polarity is positive
a_score	1	As polarity is positive, adjective score is set to 1

v = ‘working’
nw = ‘not’
v_score = 0 + 1
v_score = 1

This concept is shown in step 1 of the findscore function algorithm.

9. Calculate the final negative feedback score of aspect ‘display’ and store it in the CURSCORE table (shown in step 2 of findscore function algorithm).
Output:

score = n_score + a_score + v_score
score = 0 + 11 + 1
score = 12

Store value of score in CURSCORE table for the aspect ‘display’.

6.1.4 Display negative feedback score

1. To print the negative feedback score of each aspect of the entity, the showscore function is called (shown in step 1 of the showscore function algorithm).
Output:

The negative feedback score of the aspect ‘display’ of the entity ‘laptop’ is 12.

Similarly, different types of sentences are taken as input from the SA database and a negative feedback score is calculated for each aspect of the entity under analysis using the SentiNeg algorithm.

6.2 Results of SentiNeg Algorithm

The SentiNeg algorithm was implemented using Python programming language. The laptop review data was extracted from the e-commerce website www.amazon.com was used as input to the proposed SentiNeg algorithm.

The results after applying the SentiNeg algorithm on the laptop review dataset obtained from the e-commerce website www.amazon.com are shown in table 11. Visualization tools like tables and charts are used to show the scores of different aspects of the entity under analysis. The overall picture of each aspect of the entity and its usage is shown in Fig 2 and Fig 3. The status and feedback about the parts and different aspects of the laptop and the scores, stored in the database after the score calculation are displayed. Results of the SentiNeg algorithm include a negative feedback score of each aspect.

Table 22 shows the list of aspects extracted from the dataset for which the negative feedback score is calculated.

Table 23 shows the negative feedback score obtained for each aspect of the entity. The graphical representation of the aspects and score is shown in Figure 4 and Figure 5.

6.3 Comparison of SentiNeg Algorithm with Existing Methods/Techniques/Algorithms

Table 24 shows the comparison of the functionalities of existing methods/techniques/algorithms with the improvisations developed in the SentiNeg algorithm.

Table 22. Extracted aspects from dataset

	Extracted Aspects from dataset
1	Battery
2	Weight
3	Battery-life
4	Display
5	Brightness
6	Usability
7	Screen-Clarity
8	CD ROM
9	Bluetooth
10	Durability
11	Compatibility
12	Storage

Table 23. Negative feedback score of each aspect

Aspect	Score
Battery	720
Weight	645
Battery-life	550
Display	810
Brightness	940
Usability	430
Screen-Clarity	850
CD ROM	540
Bluetooth	780
Durability	820
Compatibility	580
Storage	750

7. CONCLUSION

The data used to perform sentiment analysis is available on the web in both structured and unstructured form. It is important to understand the negative part of the data because it gives the scope of improvement in the respective domain. Negation analysis, being one of the important tasks of sentiment analysis, is complex. There are different linguistic and grammatical rules applicable to perform negation analysis. In this paper, negation rules applicable to different negation words and negation scope detection are proposed. A negation handling algorithm called SentiNeg has been proposed for sentence-level sentiment analysis, which performs negation handling on opinionated sentences. The proposed SentiNeg algorithm filters non-opinionated sentences from the data to avoid unnecessary

Figure 4. Aspects and feedback score of each aspect of the laptop

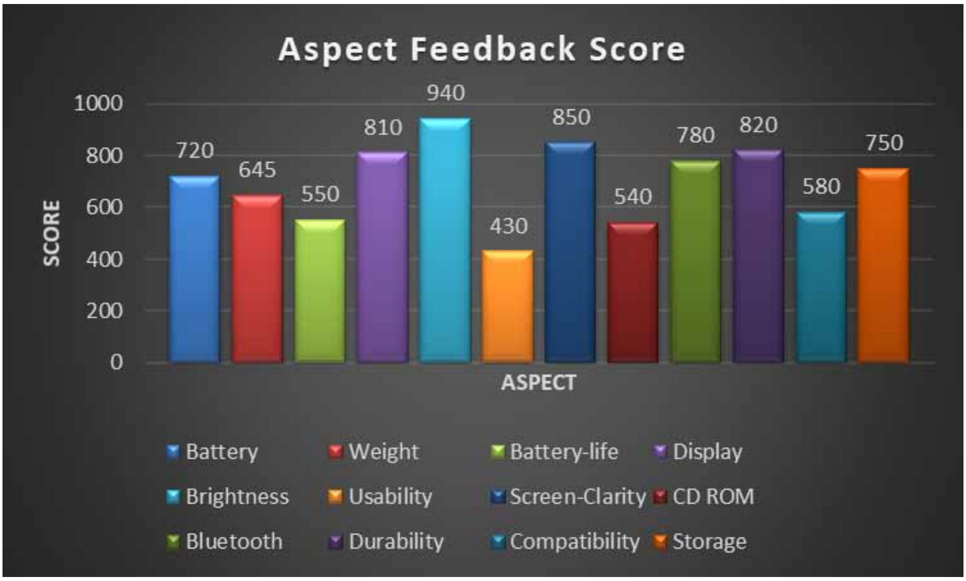
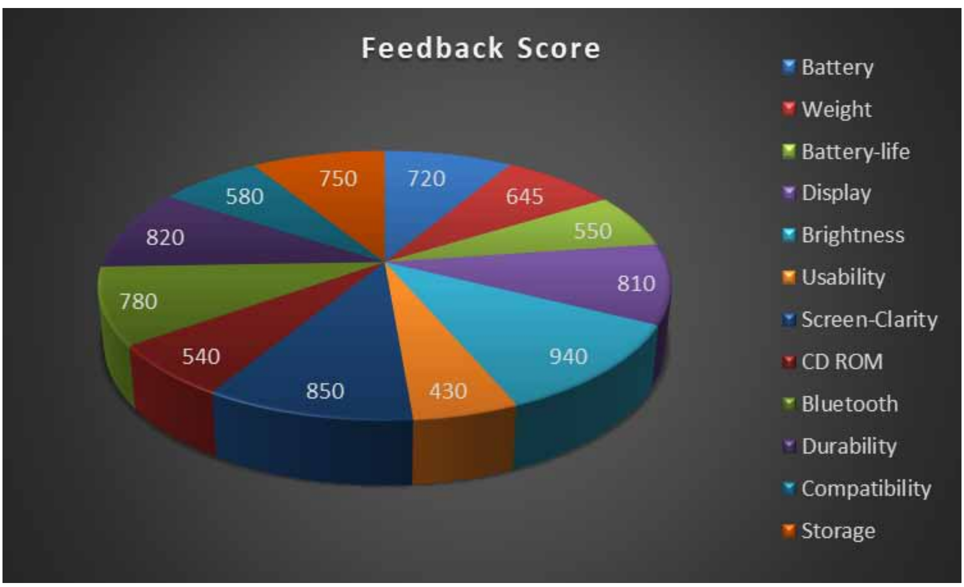


Figure 5. Aspects and feedback score of each aspect of the laptop



processing. For the opinionated sentences, SentiNeg considers different linguistic or grammatical rules of the language, that apply to the proposed categories of negation words used in the sentence. The proposed SentiNeg algorithm also gives a detailed aspect-based summary of negative opinions expressed on the entity under analysis.

Table 24. Comparison of SentiNeg algorithm with existing methods/techniques/algorithms

Technique/Method/ Concept/Algorithm	Existing Shortcomings	Improvisations implemented in SentiNeg Algorithm
Bag-of-Words (BOW)	Discards the word order and hence results in the loss of semantic information of the text, resulting in polarity shift problem.	Addresses polarity shift problem by using the extensive linguistic rules applicable to each negation word.
Algorithm based on Dependency Parse Tree	Ignores the influence of conjunctions and punctuation marks.	Considers conjunctions and punctuation marks by using rules applicable in identifying the scope of negation.
Naïve Bayes Algorithm	Used for sentiment classification into positive, negative or neutral.	Performs negation analysis using different linguistic and grammatical rules.
Support Vector Machine Algorithm	Used for sentiment classification into positive, negative or neutral.	Performs negation analysis using different linguistic and grammatical rules.
Handling Approximate Negators	Absolute negators like 'not', 'never' etc are being considered and approximate negators are not concentrated. Hence it poses a challenge in negation scope detection.	Considers approximate negators for negation analysis.
Handling downtoners	Results are not accurate as downtoners reduce the intensity of accompanying words rather than inverting the polarity.	Analyses downtoners during negation analysis and assigns partial negation score rather than declaring it as completely negative.
Detecting scope of Negation	The cases where there is no scope for negation are considered for negation analysis.	Identifies scope of negation before performing negation analysis.
Handling adjectives/verbs with negative prefix/suffix	Adjectives/verbs with negative prefix/suffix need more attention and are misinterpreted.	Adjectives/verbs with negative prefix/ suffix are split and validated to identify their polarity.

8. FUTURE ENHANCEMENTS

In this work, we have concentrated on different negation rules to perform negation analysis. In future work, sentences using different punctuation marks like single quotes, double quotes, a hyphen, dashes, etc can be considered for negation analysis. There are also exceptions to the punctuations considered in this paper. The sentences containing more than one conjunction also need to be considered. For all these enhancements, rules have to be defined to aid in negation analysis.

CONFLICT OF INTEREST

The authors of this publication declare there is no conflict of interest.

FUNDING AGENCY

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

Alamoodi, A H., Zaidan, B B, Zaidan, A A, Albahri, O S, Mohammed, K I, Malik, R Q, Almahdi, E M, Chyad, M A., Tareq, Z., Albahri, A S., Hameed, H., & Alaa, M. (2020). Sentiment analysis and its applications in fighting COVID-19 and infectious diseases: A systematic review. *Expert Systems With Applications*, *167*. DOI: 10.1016/j.eswa.2020.11415

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. In *Proceedings of the International Conference on Data and Information Science*. doi:10.1088/1742-6596/971/1/012039

Apostolova, E., & Velez, T. (2018). *Toward Automated Early Sepsis Alerting: Identifying Infection Patients from Nursing Notes.* arXiv:1809.03995v1 [cs.CY].

Asmi, A., & Ishaya, T. (2012). Negation Identification and Calculation in Sentiment Analysis. *Proceedings of the Second International Conference on Advances in Information Mining and Management*.

Bhatia, P., Celikkaya, B., Khalilia, M., & Colak, A. (2019). *End-to-end Joint Entity Extraction and Negation Detection for Clinical Text.* arXiv:1812.05270v2 [cs.CL].

Bhuiyan, M., Misra, A., Tripathy, S., Mahmud, J., & Akkiraju, R. (2018). Don't get Lost in Negation: An Effective Negation Handled Dialogue Acts Prediction Algorithm for Twitter Customer Service Conversations. *Proceedings of the ICWSM*, 25-28.

Brassard-Gourdeau, E., & Khoury, R. (2018). *Impact of Sentiment Detection to Recognise Toxic and Subversive Online Comments*. arXiv:1812.01704 [cs.CL].

Cheng, K., Baldwin, T., & Verspoor, K. (2017). Automatic Negation and Speculation Detection in Veterinary Clinical Text. In *Proceedings of the Australian Language Technology Association Workshop* (pp:70-78). Academic Press.

Cruz, N. P., Taboada, M., & Mitkov, R. (2016). A Machine-Learning Approach to Negation and Speculation Detection for Sentiment Analysis. *Journal of the Association for Information Science and Technology*, *67*(9), 2118–2136. doi:10.1002/asi.23533

Dalianis, H. (2018). *Clinical Text Mining*. SpringerOpen. doi:10.1109/ICECIT.2017.8453357

Diamantini, C., Mircoli, A., & Potena, D. (2016) A Negation Handling Technique for Sentiment Analysis. In *Proceedings of the International Conference on Collaboration Techniques and System (CTS)*. doi:10.1109/CTS.2016.0048

Fancellu, F., Lopez, A., & Webber, B. (2016). Neural Networks for Negation Scope Detection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (pp. 495-504). doi:10.18653/v1/P16-1047

Fancellu, F., Lopez, A., & Webber, B. (2018). *Neural networks for cross-lingual negation scope detection*. arXiv:1810.02156v1 [cs.CL].

Farooq, U., Mansoor, H., Nongaillard, A., Ouzrout, Y., & Qadir, M. A. (2017). Negation Handling in Sentiment Analysis at Sentence Level. *Journal of Computers*, *12*(5), 470–478. Advance online publication. doi:10.17706/jcp.12.5.470-478

Gaind, B., Syal, V., & Padgalwar, S. (2018). Emotion Detection and Analysis on Social Media. *Proceedings of the International Conference on Recent Trends In Computational Engineering and Technologies (ICTRCET).*

Gomez-Espinos, V., Muniz-Sanchez, V., & Lopez-Monroy, A. (2022). AutoML and Ensemble Transformers for Sentiment Analysis in Mexican Tourism Texts. *Proceedings of the CEUR Workshop*.

Gupta, I., & Joshi, M. (2021). Real-Time Twitter Corpus Labelling Using Automatic Clustering Approach. *International Journal of Computing and Digital Systems*. DOI: 10.12785/ijcds/100150

Hussein, D. M. E. M. (2018). A survey on sentiment analysis challenges. *Journal of King Saud University –. Engineering and Science*, *30*(4), 330–338. doi:10.1016/j.jksues.2016.04.002

Jimenez-Zafra, S. M., Morante, R., Martin-Valdivia, M. T., & Urena-Lopez, L. A. (2018). A review of Spanish corpora annotated with negation. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 915-924). Academic Press.

Joshi, N S., & Itkat, S A. (2014). A Survey on Feature Level Sentiment Analysis. *International Journal of Computer Science and Information Technologies*.

Kharde, V. A., & Sonawane, S. S. (2016). Sentiment Analysis of Twitter Data. *International Journal of Computers and Applications*, *139*(11).

Kiritchenko, S., & Mohammad, S. M. (2016). The effect of Negators, Modals, and Degree Adverbs on Sentiment Composition. In *Proceedings of the NAACL-HLT* (pp. 43-52). doi:10.18653/v1/W16-0410

Li, H., & Lu, W. (2018). Learning with Structured Representations for Negation Scope Extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics* (pp. 533-539). doi:10.18653/v1/P18-2085

Lincy, W., & Kumar, M. N. (2016). A Survey on Challenges in Sentiment Analysis. *International Journal of Emerging Technology in Computer Science and Electronics*, *21*(3).

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan and Claypool Publishers.

Ljajic, A., & Morovac, U. (2018). Improving Sentiment Analysis for Twitter Data by Handling Negation Rules in the Serbian Language. *Computer Science and Information Systems*, *16*(1), 289–311. doi:10.2298/CSIS180122013L

Lopes, V., Gaspar, A., Alexandre, L. & Cordiero, J. (2021). An AutoML-based Approach to Multimodal Image Sentiment Analysis. *IJCNN,* 1-9.

Mohammad, S. M. (2021). *Sentiment Analysis: Automatically Detecting Valence, Emotions, and Other Affectual States from Text*. arXiv:2005.11882v2 [cs.CL].

Ono, J., Lopez, R., & Bertini, E., & Freire. (2020). PipelineProfiler: A Visual Analytics Tool for the Exploration of AutoML Pipelines. *IEEE Transactions on Visualization & Computer Graphics*, *27* (2), 390-400. doi: doi:10.1109/TVCG.2020.3030361

Palomino, N. E. (2018). *The role of approximate negators in modeling the automatic detection of negation in tweets* [Unpublished doctoral dissertation]. Syracuse University.

Peng, Y., Wang, X., Lu, L., Bagheri, M., Summer, R., & Lu, Z. (2018). NegBio: a high performance tool for negation and uncertainty detection in radiology reports. In *Proceedings of the AMIA 2018 Informatics Summit* (pp. 188-196). AMIA.

Raheem, M. (2020). Sentiment Analysis in Healthcare: Motives, Challenges & Opprtunities Pertaining to Machine Learning. *International Journal of Management*, *11*(11), 1166–1174. doi:10.34218/IJM.11.11.2020.109

Sarkar, K. (2020). Heterogeneous classifier ensemble for sentiment analysis of Bengali and Hindi tweets. *Sadhana. Indian Academy of Sciences*, *45*, 196. doi:10.1007/s12046-020-01424-z

Savanur, S. R., & Sumathi, R. (2017). Feature Based Sentiment Analysis of Compound Sentences. *Proceedings of the 2nd International Conference on Emerging Computation and Information Technologies (ICECIT).*

Schulder, M., Wiegand, M., & Ruppenhofer, J. (2018). Automatically Creating a Lexicon of Verbal Polarity Shifters: Mono- and Cross-lingual Methods for German. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 2516-2528). Academic Press.

Siriborvornratanakul, T. (2022). Human behavior in image-based Road Health Inspection Systems despite the emerging AutoML. *Journal of Big Data*, *9*, 96. doi:10.1186/s40537-022-00646-8

Strohm, F., & Kilnger, R. (2018). An Empirical Analysis of the Role of Amplifiers, Downtoners, and Negations in Emotion Classification in Microblogs. In *Proceedings of the International Conference on Data Science and Advanced Analytics*. DOI: doi:10.1109/DSAA.2018.00087

Taboada, M. (2015). Sentiment Analysis: An Overview from Linguistics. A Review. *Annual Review of Linguistics*, *2*(1), 325–347. doi:10.1146/annurev-linguistics-011415-040518

Tavchioski, I., Koloski, B., Skrlz, B., & Pollak, S. (2022). E8-IJS@LT-EDI-ACL2022 - BERT, AutoML and Knowledge-graph backed Detection of Depression. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion* (pp. 251-257). Academic Press.

Triyono, S., & Sahayu, W., & Margana. (2020). Form and function of negation in German and Indonesian: Searching for equivalent construction of meaning. *Indonesian Journal of Applied Linguistics*, *9*(3), 675–684. doi:10.17509/ijal.v9i3.23218

Vo, K., Nguyen, T., Pham, D., Nguyen, M., Truong, M., Mai, T., & Quan, T. (2019). *Combination of Domain Knowledge and Deep Learning for Sentiment Analysis of Short and Informal Messages on Social Media*. arXiv:1902.06050v1 [cs.CL].

Xia, R., Xu, F., Yu, J., Qi, Y., & Cambria, E. (2016). Polarity Shift Detection, Elimination and Ensemble: A Three-Stage Model for Document-Level Sentiment Analysis. *Information Processing & Management*, *52*, 36–45. doi:10.1016/0306-4573

Yadav & Singal. (2019). New Algorithm For Product Based Sentiment Analysis and Spam. *International Journal for Technological Research in Engineering, 6*(5).

Zafra, S. M. J., Taboada, M., & Valdivia, M. T. M. (2020). Negation detection for sentiment analysis: A case study in Spanish. *Natural Language Engineering*, *27*(2), 1–24. doi:10.1017/S1351324920000376

Zamaraeva, O., Howell, K., & Rhine, A. (2018). Improving Feature Extraction for Pathology Reports with Precise Negation Scope Detection. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 3564-3575). Academic Press.

Sandhya R. Savanur is serving as Assistant Professor in the department of Computer Science and Engineering in GM Institute of Technology, Davangere, Karnataka. Her research interests include data mining, big data analysis and data analytics.