# Phrase-Based Heuristic Sentiment Analyzer for the Telugu Language

[1]Durga Prasad Manukonda, [2]Rohith Gowtham Kodali, [3]Dheeraj Guduri
ASRlytics Inc., Auxo Labs

**ABSTRACT**

This study aimed to investigate linguistics-based approach on sentiment analysis in Telugu language. Sentiment analysis is the process of classifying sentences as positive, negative, or neutral. One of the uses of this technology is to find the perceptions towards organizations and their services through their feedback in text format. With rapid growth of social media discussions on social networks, lots of data in the Telugu language is available. The new sentiment analyzer models are based on previous models made for the English language. These translate sentences with help of translation APIs into English and analyze the sentiments. One can observe compounding errors by faulty machine translation. Modern translators are inefficient are unreliable. This model is specially designed for Telugu language with newly identified rules. This model has been successfully tested for the Telugu language. The corpus for this study was gathered by using BootCa we have manually tagged root words with their polarity. We have identified and generated some important rules for identifying the sentiment in the sentence. These rules have been explained and discussed at length along with examples.

**Index Terms—** NLP, SentiWordNet, Sentiment Analysis, BootCaT, Wikipedia database

## 1. INTRODUCTION

Dravidian languages are spoken in the southern Indian states (Andhra Pradesh, Telangana, Tamil Nadu, Kerala, and Karnataka). There is a wide array of social media articles and documents available in local languages, with many social networking sites adding support for them, but there is dearth for NLP tools in these languages.

Telugu is spoken by 75 million people and is the third most spoken Indian language by the number of native speakers. Telugu ranks fifteenth on the list of most spoken languages worldwide and there are many web- sites and blogs in Telugu. To computationally identify, quantify, study and categorize opinions expressed or implied in pieces of text, we use a sentiment analyzer, especially in research to focus on opinion mining [1]. The sentiment analyzers in Telugu are English-based tools and give low accuracy [2]. They produce inaccurate re- sults most times. To overcome this problem we developed a sentiment analyzer with certain rules. A rule- based approach is one which uses rules of heuristics to determine sentiments. It uses research in linguistics to analyze sentiments. The main logic of this algorithm has been written based on truth table logic.

The article is organized as follows: In section 2, Review of literature 3, corpus-collection from different sources, data-prepossessing and creating new sentiment- word dictionaries are discussed. In section 4, Telugu grammar is discussed along with precautions needed to be taken while collecting sentiment words from Telugu language [3]. Section 5 shows methodology and discovery of new rules from Telugu grammar and algorithm design in Perl and Python are discussed in detail. Section 6 showcases the implementation of the algorithm and sample outputs are covered, Section 7 shows the conclusion and future work is discussed.

## REVIEW OF LITERATURE

The Telugu sentiment analyzers that have been studied here have improper translations using Babel Fish translator and Google translator [4]. Some of them have done.

Theoretical work on Telugu data and developed SentiWordNet. Others have done some basic classification algorithms and sentiment Classification for Telugu text using various Machine Learning techniques. None of their data was made available to the public [5]. Their approach is a combination of methodologies: effective negation handling, feature-selection by mutual information and word n- grams. This improved the accuracy. People working on Telugu depend on translators and traditional ML methods and word vectors because Telugu is a low-resource language.

Many approaches were proposed to capture the sentiment in texts; each of these approaches addressed the issue at different levels of granularity. Some researchers have proposed methods for document-level sentiment classification [6][7].

One of the authors [8] built Telugu SentiWordNet on the news corpus to perform sentiment analysis tasks. Another study developed a polarity annotated corpus, in which positive, negative and neutral words are as- signed to 5410 sentences in the corpus collected from several sources [5]. This corpus is a gold standard corpus, aimed at improving sentiment analysis in Telugu. In order to minimize the dependence of ML approaches for sentiment analysis on abundance of corpus, this pa- per proposes a novel method to learn representations of resource-poor languages by training them jointly with resource-rich languages using a Siamese network, a novel approach to classify sentences into their corresponding sentiment using contrastive learning, which uses shared parameters of Siamese networks [9].

Sentiment expressed about a particular entity at the top level of granularity, as document may convey different opinions for different entities. When we consider the tasks of opinion mining, where the aim is to capture the sentiment polarities about the entities, such as products in product reviews, it is shown that sentence-level and phrase-level analysis leads to a performance gain, seen [10], [11] propose an alternate way, in the context of Indian languages, to build the resources for multilingual effect analysis where translations into Telugu are done using WorldNet. Only two reported works exist in Telugu sentiment analysis using sentence-level annotations, that developed annotated corpora [5][12]. This approach is first of its kind in NLP research which uses the rules observed at phrase level of the sentence for opinion mining and sentiment analysis.

In the section below, we talk about some methodologies and approaches used in addressing the task of sentiment analysis and polarity classification. Our work derives inspiration from most of this work.

**CORPUS COLLECTION, DATA PREPROCESSING AND DEVELOPMENT OF SENTIWORDS**

For the collection of data we used Telugu Wikipedia database dump and extracted text from it using Wikiex- tractor.py, https://github.com/attardi/wikiextractor. Further, the data was preprocessed using Python-based text-cleaning scripts. With some amount of manual work top 20000 most frequent words were generated and tagged with their respective Parts-of-Speech tags. We then manually identified verbs carrying Positive and Negative sentiments, ending up with 1335 negative words and 1051 positive words.

Synonyms of these words were taken from grammar books like Shabda Ratnakaram [13]. Text was generated using BootCaT using these words as seed words. We used the output as our dataset. [14] And from this dataset we further derived these rules which qualify a sentence as positive or negative.

## 1. TELUGU GRAMMAR

Telugu is a Dravidian language and is more inflected than other literary Dravidian languages and the general structure of a Telugu sentence is in the order Subject- Object—Complement-Verb(S-O—C-V). The grammar of Telugu is quite similar to Kannada, which is one of the Dravidian languages. The sentence ends with a verb+PNG (Person-Number-Gender) or a verb. Most of the rules depend on verbs (Positive or Negative verbs (V) or compound verbs (main verb + auxiliary verb)) and Adjectives (JJ) or Complements. An important point regarding a sentiment analyzer is that it's tough to de- cide the sentiment or polarity of the sentence based on a single wor, it depends on the current, previous and following words of the sentence. For example:

Ramu bhadagā unnaḍu (S-O-V)
Ramu is sad

Ramu bhadagā leḍu (S-O-V)
Ramu is not sad

After observing the above sentence, it was noticed that the sentiment of an English sentence depends on the single negative word not, in Telugu it depends on the current, next and previous words. Verbs unnaḍu is in positive form and leḍu is in negative form and bha da(gā) means sadness. So, the sentiment Telugu sentence with the negative word bhada(gā) also depends on verbs leḍu or unnā .u. So when the current word is negative, it depends on the next verb (negative or positive).

Telugu is a comparatively highly Sanskritized language among Dravidian languages and uses morphological processes to join words together, forming complex words. These processes are traditionally referred to as Sandhi. Example, nara + indra gives the word Narendra. Coming to verbs, the complex form of a verb is in the Verb + Tense + PNG (Person Number Gender). Exam- ple, unnā ḍu (Verb = unna, Tense = Past, PNG = ḍu). Sowe have to take more precautions while collecting sen-timent words. Negative complex word formation is one of the main features of negative and positive complex words in Telugu:

Negative complex-word formation

verb/adverb + negative auxiliary verb = negative word

When the complex word has the form verb + neg- ative auxiliary verb, the result is a negative word and same is true for nouns and adverbs occurring before the auxiliary verb. These are examples:

avvalē du "did not happen" (avva+le du, verb = avva, auxiliary verb = ledu)

cē yalē du "did not do" (cē ya+lē du, verb = cē ya, auxiliary verb = le )

Positive complex-word formation verb/adverb + positive auxiliary verb
= positive word

When the complex word has the form verb + positive auxiliary verb, the result is a positive word and same is true for nouns and adverbs occurring before the auxiliary verb. These are examples:

- tiṇṭunnā nu "am eating" (tiṇṭu+unnā nu, verb = tiṇṭu, auxiliary verb = unnā nu)
- bā gunnā ḍu "He is looking good" (bā gā +unnā ḍu, verb = bā gā , auxiliary verb = unnā ḍu)

Vowel-length based Polarity Words change polarity (positive and negative) with little difference in ending of verb like:
/a/ a /a/ a¯
[16] considering Positive=POS, Negative=NEG, NEU=
Neutral

nacca nu (POS); naccanu (NEG)
"Am liked" (POS); "am not liked" (NEG)

sammatinca nu (POS); sammatincanu (NEG) "Iagreed"(POS);"Ididnotagree"(NEG)

A few other direct sentiwords occur apart from the above rules. Also, if the prefix of few words is or apa, the polarity changes to negative.

Some researchers proposed computational techniques to generate sentiment lexicons in Indian languages, including Bengali, Hindi, and Telugu, automatically or semi-automatically. By observation and further research on the Telugu language, the polarity is found to not depend on a single word, but a phrase or multiple phrases of the sentence. This has been explained in the sections below. This can be one of the reasons why most sentiment analyzers in Telugu arent working properly. To overcome this problem, we have proposed a new method called Phrase based Heuristic Sentiment Analyzer.

## 2. DEVELOPMENT OF ALGORITHM

For the development of sentiment analysis, a phrase- based approach has been taken. Processing of the text has been done initially using NLP tools SpaCy, NLTK or Stanford NLP, to tokenize and tag the sentences. After assign the lexicon weighting the polarity of individual words, SpaCy is used to check if the next word is a verb. This process is continued and if the next word is a verb, phrase-based polarity classification is done. The algorithm design at this level of processing is explained in the next subsection. If the next word is not a verb, the process goes to the final level of average calculation of word polarities. The averages score of the complete sentence is calculated based on below average formula:

$$\text{Average Score} = \frac{\text{Sum of polarities}}{\text{Total number of polarities after phrase level operations}}$$

The average score lies in between -1 to +1. The sentiment deciding factors are:

- 0 = neutral (NEUT)
- 0.1 to 1 = positive (POS)
- -0.1 to -1 = negative (NEG)

The sentiment is derived here based on average range.

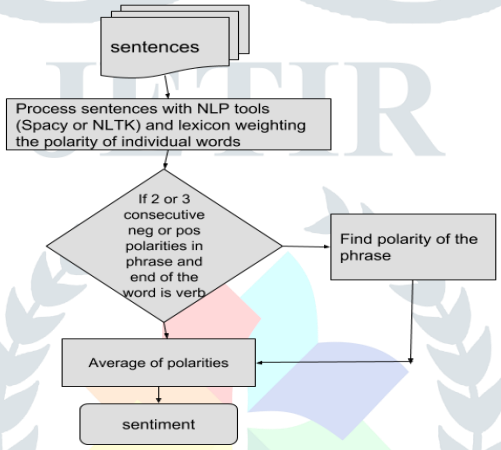


Fig. 1. Describing the Basic Process in Phrase based Heuristic Sentiment Analysis

### Rules for Algorithm Development

In polarity calculation at phrase level, we describe the programming rules for algorithm. With most heuristic models, the sentiment of the phrase is derived by using a truth table and few programming rules. There are nine such rules that were employed for the algorithm design:

**Rule 1: POS NEG = NEG**

POS NEG (VERB + AUXVERB OR AUXVERB) = NEG
1 -1 = -1

paṭṭinchukovaḍam (+1) lēdu (-1) = -1 “does not care”(Negative Statement)

iṣṭam (+1) lēdu (-1) = -1 “does not like” (Negative Statement)

**Rule 2: NEG POS = NEG**

NEG POS (VERB + AUXVERB OR AUXVERB) = NEG
-1 1 = -1

narakamgā (-1) undi (+1) = -1 “it is like hell”
(Negative Statement)

bhayam (-1) undi (+1) = -1 “there is fear”
(Negative Statement)

**Rule 3: NEG NEG = POS**

NEG NEG (VERB + AUXVERB OR AUXVERB) = POS
-1 -1 = 1

raddu (-1) lēdu (-1) = +1 “there is no abolishment” (Positive Statement)

kaṣṭam (-1) lē du (-1) = +1 “there is no difficulty” (Positive Statement)

**Rule 4: POS POS = POS**

POS POS (VERB + AUXVERB OR AUXVERB) = POS
1 1 = 1

iṣṭam (+1) undi (+1) = +1 “does like”
(Positive Statement)

santō ṣamgā (+1) undi (+1) = +1 “I feel happy”
(Positive Statement)

**Rule 5: NEG NEG POS = NEG**

NEG NEG POS (VERB + AUXVERB OR AUXVERB) = NEG
-1 -1 1 = -1

bhayam (-1) bhayamgā (-1) undi (+1) = -1 “I feel so so scared”
(Negative Statement)

kō pamgā (-1) chirā kugā (-1) undi (+1) = -1 “I am angry and frustrated”
(Negative Statement)

**Rule 6: NEG NEG NEG = POS or NEG**

Rule 6(a): NEG NEG NEG (AUXVERB) = POS

NEG NEG NEG (AUXVERB) = POS
-1 -1 -1 = +1

bhayam (-1) bhayamgā (-1) lē du (-1) = +1 “I don’t feel so scared”
(Positive Statement)

bā dha (-1) bā dhagā (-1) lē du (-1) = +1 “I don’t feel so sad”
(Positive Statement)

Rule 6(b): NEG NEG NEG (VERB + AUXVERB) NEG NEG NEG (VERB + AUXVERB) = NEG
-1 -1 -1 = -1

nariki (-1) nariki (-1) campā ḍu (-1) = -1 “By cutting and cutting, he killed it” (Negative Statement)

**Rule 7: POS POS NEG = NEG**

POS POS NEG (VERB + AUXVERB OR AUXVERB)
= NEG 1 1 -1 = -1

premanu (+1) premagā (+1) ivvalē du (-1) = -1 “did not show love as love”
(Negative Statement)

**Rule 8: POS NEG NEG = POS**

POS NEG NEG (VERB + AUXVERB OR AUXVERB) = POS
1 -1 -1 = 1

nidhulu(+1) durviniyō gam (-1) avvalē du (-1) = +1 “funds did not get misused”
(Positive Statement)

Rule 9: Conjugation using kā ni between two polarities

This rule is applicable only when there is a conjugation of two polarities with the word kā ni. The following ex- amples help further:

nā ku iṣṭamgā undi kā ni andamgā lē du “I like it but I don’t feel it’s pretty”
0 +1 +1 0 +1 -1 = 0 +1 0 -1 = 0 +1 = +1
(Positive Statement)

nā ku bā dhagā undi kā ni narakamgā lē du “I feel sad but I don’t feel horrible”
0 -1 +1 0 -1 -1 = 0 -1 0 + 1 = 0 -1 = -1
(Negative Statement)

## 3. EXPERIMENTS AND RESULTS

A sample set of 6011 Telugu sentences were taken from various Telugu news websites, and sentiments were manually derived for each of them. For each word, polarity was derived from the dictionary that we gener- ated, using Trie-based searching algorithms. The rules were then applied to obtain an accuracy of 92%. The accuracy can be increased further in the future, and that needs an extensive dictionary.

Negative words have been assigned -1, positive words with +1 and neutral words have been assigned with 0. In Python, Trie implementation was used [17] for assigning the word sentiment and the operations were later performed based on the truth table logic shown in below Table 1 and 2.

An example sentence after assigning the sentiments is:

ī panikastamgā undi “this work is difficult” 0 0 -1 1

The Table 2 describes the sentiment operation of the two consecutive words based on their current, next state of words and this rules explained in section 5.

Table 1 describes the three consecutive words based on their previous, current and next states of the words. In rule 6(a) and 6(b), the overall polarity depends on next state of the wor If next state of the word is negative verb(ledu), the overall polarity of the three words is neg- ative or else it is positive. Examples shown in section 5 thats rule 6(a) and 6(b).

| Sentence | English Translation | Sentiment | Sum | Average |
|---|---|---|---|---|
| nāku bādhagā undi kāni narakamgā lēdu | I am feeling bad but not horrible | NEG | -1 | -0.5 |
| ī pani kaṣṭamgā undi | This work is difficult I | NEG | -1 | -0.5 |
| nāku santōṣamgā undi | am feeling happy | POS | 1 | 1.0 |
| sūryuḍu dakṣiṇāna udayistaḍu | The sun rises in the south | NEU | 0 | 0.0 |
| sinimā manchigā bāgundi | The movie is good | POS | 1 | 0.5 |
| nāku bhayam bhayamgā undi | I am feeling so so scared | NEG | -1 | -1 |

Table 1. Result table for various sample sentences.

nāku santōṣamgā undi “I feel happy”

0 1 1

0 1

| Rule | Current word | Next word | Result |
|---|---|---|---|
| 1 | +1 | -1 | -1 |
| 2 | -1 | +1 | -1 |
| 3 | -1 | -1 | +1 |
| 4 | +1 | +1 | +1 |

**Table 2**. Derived truth table for rules 5.1 to 5.4

| Rule | Previous | Current | Next | Result |
|---|---|---|---|---|
| 5 | -1 | -1 | +1 | -1 |
| 6a | -1 | -1 | -1 | +1 |
| 6b | -1 | -1 | -1 | -1 |
| 7 | +1 | +1 | -1 | -1 |
| 8 | +1 | -1 | -1 | +1 |

**Table 3**. Derived truth table for rules 5.5 to 5.8

A few computer generated table operations for sam- ple sentences:

ī pani kaṣṭamgā undi “this work is difficult” 0 0 -1 1

0 0 -1

Total Polarites = 3 Sum = -1

nāku bādhagā undi kāni narakamgā lēdu “I feel sad but I don’t feel horrible”

0 -1 1 0 -1 -1

0 -1 0 1

0 -1

Total Polarites = 2 Sum = -1

Total Polarites = 2 Sum = 1

nariki nariki champa du

"By cutting and cutting, he killed it"

-1 -1 -1

-1

Total Polarites = 1 Sum = -1

Finally the stop words were deleted for accuracy.

The sample is shown in Table 1.

| Sentiment Analysers | Total sentences | Accuracy |
|---|---|---|
| ML-Linear Regression | 5410 | 68.17% |
| ML-Naive Bayes | 5410 | 64.85% |
| ML-Random Forest | 5410 | 66.55% |
| SSA | 6011 | 29.77% |
| PBHSA | 6011 | 92.00% |

Table 4. Accuracy comparison table.

Phrase based Heuristic Sentiment Analyzer (PBHSA) was compared with Enhanced Sentiment Classification of Telugu Text using Machine Learning Techniques [18] and Google's translation engine with Stanford Sentiment Analyzer (SSA). Using the same data set, we see that the sentences that are translated to English by using Google's API with Stanford Sentiment Analyzer(SSA) give an accuracy of 29.77% and with a different data set, the highest accuracy got by using the ML techniques is 68.17%. Thus there is an increase in accuracy by 23.83%.

## 4. CONCLUSION AND FUTURE WORK

The results are insightful, considering the fact that Telugu is an agglutinative language. Sentiment analysis so far has never been done on agglutinative Dravidian languages. Since our work is the first attempt of this kind, the sentiment analyzer that we have built using a Phrase- based Heuristic approach seems to be working well in determining the sentiments of sentences properly. The individual tests show the accuracy falling at 80-94%.

Work is in progress for other Dravidian Languages (Tamil, Kannada and Malayalam).

## 5. REFERENCES


[1] Awais Athar, "Sentiment analysis of citations us- ing sentence structure-based features," in Proceed- ings of the ACL 2011 Student Session, Stroudsburg, PA, USA, 2011, HLT-SS '11, pp. 81–87, Associa- tion for Computational Linguistics.

[2] N.J. Khan, Waqas Anwar, and Nadir Durrani, "Machine translation approaches and survey for in- dian languages," 2017.

[3] Alejandro Gutman and Beatriz Avanzati, "Telugu," 2013.

[4] Gayane Shalunts, Gerhard Backfried, and Nicolas Commeignes, "The impact of machine translation on sentiment analysis," 2016.

[5] Sandeep Sricharan Mukku and Radhika Mamidi, "Actsa: Annotated corpus for telugu sentiment analysis," in Proceedings of the First Workshop on Building Linguistically Generalizable NLP Sys- tems. 2017, pp. 54–58, Association for Computa- tional Linguistics.

[6] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan, "Thumbs up?: Sentiment classification using machine learning techniques," in Proceedings of the ACL-02 Conference on Em- pirical Methods in Natural Language Processing - Volume 10, Stroudsburg, PA, USA, 2002, EMNLP '02, pp. 79–86, Association for Computational Linguistics.

[7] Peter D. Turney and Michael L. Littman, "Measur- ing praise and criticism: Inference of semantic ori- entation from association," ACM Trans. Inf. Syst., vol. 21, no. 4, pp. 315–346, Oct. 2003.

[8] R. Naidu, S. K. Bharti, K. S. Babu, and R. K. Mo- hapatra, "Sentiment analysis using telugu senti- wordnet," in 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), March 2017, pp. 666–670.

[9] Nurendra Choudhary, Rajat Singh, Ishita Bindlish, and Manish Shrivastava, "Emotions are univer- sal: Learning sentiment based representations of resource-poor languages using siamese networks," CoRR, vol. abs/1804.00805, 2018.

[10] Theresa Wilson, Janyce Wiebe, and Paul Hoff- mann, "Recognizing contextual polarity in phrase- level sentiment analysis," in Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Process- ing, Stroudsburg, PA, USA, 2005, HLT '05, pp. 347–354, Association for Computational Linguis- tics.

[11] Yoonjung Choi, Lingjia Deng, and Janyce Wiebe, "Lexical acquisition for opinion inference: A sense-level lexicon of benefactive and malefactive events," 01 2014, pp. 107–112.

[12] Sreekavitha Parupalli, Vijjini Anvesh Rao, and Radhika Mamidi, "Bcsat : A benchmark corpus for sentiment analysis in telugu using word-level annotations," in Proceedings of ACL 2018, Stu- dent Research Workshop. 2018, pp. 99–104, Asso- ciation for Computational Linguistics.

[13] B. Sitaramacharyulu, Sabda ratnakaram: a dictio- nary of Telugu language, Asian Educational Ser- vices, 1885.

[14] Marco Baroni and Silvia Bernardini, "Bootcat: Bootstrapping corpora and terms from the web,"
.

[15] ISO-TC-46, Information and documentation Transliteration of Devanagari and related Indic scripts into Latin characters, ISO 15919:2001, 2001.

[16] Communications & Information Technology Min- istry Govt. of India, "Unified parts of speech (pos) standard in indian languages," 2007.

[17] Phil Bagwell, "Fast and space efficient trie searches," 06 2000.