Analyzing the Emotions in Telugu

Mr. Ch. Appalanaidu, Mr. B.V.PhaniRaju

Mr. B. Gowrishankar (Corresponding Author)

GAYATRI COLLEGE OF SCIENCE & MANAGEMENT Accredited with by NAAC & ISO

(Affiliated to Dr. B R AMBEDKAR UNIVERSITY)

Under the Management of GURAJADA EDUCATIONAL SOCIETY

Munasabpeta, SRIKAKULAM – 532 401

Abstract

Many NLP activities, including recommendation systems, question answering, and business intelligence products, now rely on sentiment analysis as a major component. An extract is analyzed to determine if it is mostly positive or negative in terms of sentiment analysis, which is essentially an exercise of looking for emotional content in a piece of writing. There has been a lot of interest in sentiment analysis in English, but it hasn't had the same impact in Telugu. Sentimental analysis for the language described as "the Italian of East" is examined in this thesis. Despite the difficulties of sentiment analysis, the lack of appropriate data in Telugu has stifled research efforts in the language. As a result, we begin by addressing the latter issue in order to better handle the former. We initially manually mark Telugu texts with a positive, negative, or neutral tag in order to solve the issue of data availability. A number of machine learning methods are then used to the annotated data in order to tackle sentiment analysis classification issues that may be either binary (only positive or negative tags) or ternary (positive, neutral and negative tags). Even though Random Forests are the most complicated classifier for binary classification tasks, logistic regression may outperform them when the challenge is presented as three-tiered classification. In general, as the quantity of training data increases, so does the performance of machine-learning algorithms. Manual annotation, on the other hand, is time consuming and costly. Hybrid Query Selection Strategy is used to overcome these concerns by using an annotated data set that includes a variety of different classifiers. Experiments have shown that our method accomplishes this goal with a low level of inaccuracy. Classifying phrases based on their polarity is another way we employ active learning to achieve our final aim.

Introduction

Introduction to Sentiment Analysis

Analyzing a speaker's or writer's sentiments, emotions, and attitude in a particular text is the focus of sentiment analysis. In order to find and extract subjective information from source materials, sentiment analysis or opinion mining makes use of techniques such as natural language processing, computational linguistics, and text analytics. Web-generated material may be used for sentiment analysis, which focuses on determining the persona of a user based on the content they provide. There are several commercial and academic uses for this growing sub-discipline at the intersection of Natural Language Processing and Information Retrieval. Using this data, you may learn more about the user's preferences, likes, and dislikes. The words "sentiment analysis" and "opinion mining" are sometimes used interchangeably in the scientific community.

Content Created by Others

In recent years, with the advent of the Internet, there has been an exponential rise in user-generated content (UGC). People who are eager to share knowledge, data, and media over the internet in different formats, such as product/movie and restaurant reviews and ratings, wikis and blogs, news, discussion forums and so on

Testimonials and scores

The entertainment and e-commerce sectors are two of the world's biggest and fastest-growing businesses. Consumers have been able to express their opinions and sentiments about items and services through online reviews and ratings. These aid other buyers in making the best options and selections possible.

Blogs and wikis are a two-way street.

People may become content producers rather than content consumers thanks to blogs. The two most common types of blogs are topical commentary and personal online diaries. They are generally written,

but there are various formats, such as video blogs, art blogs, and so on, that are also available. Currently, there are more than 300 million publicly accessible blogs throughout the globe.

Forums of Dialogue

Online discussion forums are places where individuals get together to have text-based talks with one another. A user posts a question or request for information, and other users respond with an answer or an opinion based on what they know about the subject matter.

News

Online news sources report on current events and other noteworthy occurrences from across the world, and they do so in a variety of languages. Since the rise of user-generated content in recent years, it has become more popular due to its low cost (often free) and its usefulness in gauging a product or service's popularity/reviewability. It allows customers access to data that hasn't been tampered with by middlemen. It also opens up a slew of new possibilities for corporations and academics alike.

Telugu Content on the Web

Telugu, a Dravidian language, is spoken mostly in the country where it originated. It is the third most widely spoken language in India, according to the Ethnologue2 list of the most widely spoken languages in the world3. There has been a dramatic rise in the number of Telugu-language web sites with the adoption of Unicode (UTF-8) standards for Indian languages. Telugu-language news and entertainment websites like eenadu.net, andhrajyothy.com, and telugu.oneindia.com, as well as social media sites like Twitter, Facebook, and Instagram are all places where you may get information in Telugu. An ever-increasing number of individuals are taking advantage of the Internet's accessibility and expanding their creative output on its pages as a result. It is necessary to correctly mine this large amount of data in order to assist the public and enhance the services that are already given. Table 1.1 provides a ranking of prominent Telugu websites based on the number of visitors4.

Table 1.1 List of Popular Telugu Websites based on number of viewers

Website	Number of Viewers
Eenadu.net	2.5 million
Sakshi.com	1.8 million
Andhrajyothy.com	326,000
Telugu.oneindia.com	297,000
Teluguone.com	98,000
123telugu.com	96,300

Motivation

Humans have a built-in capacity to recognize and assess the feelings and thoughts of others. An key and fundamental issue is, however, how effectively a computer can be educated to display this phenomena. It is possible to get insight into a user's emotions, attitudes, preferences, and behavior using sentiment analysis. Sentiment analysis is a beneficial tool for both manufacturers and consumers because of the vast amount of data accessible on the Internet. Labeling the massive amount of online reviews, comments, blogs, and forums with a sentiment would be helpful to customers since it would give a concise summary for them and aid in their decision-making process. Customers' evaluations may be used to uncover patterns and enhance goods and services from a producer's perspective, using sentiment analysis. The English language is the primary focus of most sentiment analysis research. In recent years, the availability of data in regional languages, such as Telugu, has risen sharply because of the Internet's widespread accessibility. However, unlike English, the datasets and tools for sentiment analysis in Telugu are quite restricted. Sentiment analysis of the Telugu language has been my Research Problem of choice because of these factors.

Applications

Sentiment Analysis is mainly used for understanding the subjectivity of the text. We discuss few domains where sentiment analysis can be applied –

Recommendation Engines

Online music platforms and e-commerce websites alike have included features that provide users with suggestions for new music to listen to or products to buy. Sentiment analysis may be used to find out what music or things a user likes by analyzing their remarks. This method may be used as an alternative to collaborative filtering or in conjunction with it to

develop improved recommender systems that perform better.

Assist in the Making of Decisions

Choosing among the various possibilities is an essential part of everyone's life. When making decisions about what to purchase, watch, or eat, sentiment analysis may play a more important role than ever before. Emotional intelligence aids in making more informed decisions.

Infrared detection of fire

Emails and other forms of social networking are becoming more laced with profanity and other derogatory words. Subjectivity identification and sentiment analysis are two approaches that may be used when mining this kind of content.

Inquiries and Answers

Opinion-based inquiries need a different strategy. Answers that incorporate additional information regarding the subject of the question may be provided using sentiment analysis, which may play a critical part in solving these kinds of queries.

Intelligence in the workplace

When it comes to developing business strategy, a lot of product and service-based organizations follow public response. However, it is impossible to conduct an individual poll of the product's consumers. In this way, sentiment research may help in the development of a product or service that better meets the demands of the market.

Political Context

An enormous undertaking is selecting a public official, whether he or she the country's president or prime minister. In social media, people voice their opinions about the candidates. With the aid of sentiment analysis, political parties may receive a clear image of the upcoming elections and make the necessary modifications to their campaigns.

Problem Statement

This study focuses on the analysis of Telugu language emotions. For sentiment analysis in Telugu, there are few resources and datasets available. Annotating Telugu sentences using a set of rules and employing several Machine learning approaches to categorize the polarity of Telugu phrases into binary or ternary categories is the work done. In order to address the issue of limited annotated datasets, we apply active learning and propose an approach termed hybrid query selection strategy. We use three distinct classifiers, support vector machines (SVM), extreme gradient boosting (XGBoost), and gradient boosted trees, to construct a sentiment analysis model for the Telugu language via active learning (GBT).

This Thesis has made a contribution.

Telugu is the primary focus of our study in this project. The paucity of tools and datasets for sentiment analysis in the Telugu language has been previously mentioned. In this study, we created a gold-standard corpus of Telugu phrases annotated by hand that may be used by anybody. We used a variety of machine learning techniques to classify the polarity of Telugu words, including Naive Bayes, Logistic Regression, SVM (Support Vector Machines), MLP (Multi Layer Perceptron) Neural Network, Decision Trees, and Random Forest. We developed models for categorizing sentiment into positive and negative polarities and positive, negative, and neutral polarities using binary and ternary classification tasks, respectively. Comparing our human-annotated corpus with automatic classification results demonstrates its reliability. We studied the applicability of active learning for labelingunlabeled Telugu phrases with little labelled data since the human annotation procedure is time-consuming, expensive, and resource-intensive. We developed a word embedding model for the Telugu language and studied several embedding locations. Rather of relying only on labeled data, we developed a hybrid query selection mechanism for active learning. We used this strategy for Telugu and labeled a large number of previously unlabeled data points. Using support vector machines (SVM), extreme gradient boosting (XGBoost), and gradient-boosted trees as classifiers, we created a sentiment analysis model for Telugu using active learning (GBT). With a low mistake rate, we were able to get encouraging findings.

Thesis Synopsis and Schema

Rest of the thesis follows this format: In the second chapter, we review the many methodologies, degrees, and genres of sentiment analysis that have been used in the past. According to Chapter 3, we'll discuss the generation and annotation of data resources for this study. The challenge of sentiment analysis in Telugu is addressed in Chapter 4 utilizing machine learning methods. For coping with the lack of annotated data, we offer a novel hybrid query selection technique in active learning in Chapter 5. In Chapter 6, the findings and conclusions of this study are summarized and discussed.

Related Work

Rest of the thesis follows this format: In the second chapter, we review the many methodologies, degrees, and genres of sentiment analysis that have been used in the past. According to Chapter 3, we'll discuss the generation and annotation of data resources for this study. The challenge of sentiment analysis in Telugu is addressed in Chapter 4 utilizing machine learning methods. For coping with the lack of annotated data, we offer a novel hybrid query selection technique in active learning in Chapter 5. In Chapter 6, the findings and conclusions of this study are summarized and discussed.

Sentiment Analysis Methods

Earlier, several ways and procedures have been created by the community to determine the sentiment polarity of the provided text. The most frequent and extensively utilized methods are discussed below.

Syntactic Approach

Sentiment analysis utilizing N-Grams was carried out by Bo Pang, Lillian Lee, and Vaithyanathan [62]. As features, they employed standard n-grams and POS information as well as machine learning classifiers such as Naive Bayes Classification, Maximum Entropy, and Support Vector Machines to analyze sentiment. In addition, they experimented with other combinations of N-Gram characteristics, such as only the existence of unigrams or unigrams with a high frequency, bigrams, and unigrams+bigrams. They came to the conclusion that the existence of a unigrams technique combined with SVM results in the highest accuracy.

An Approach Based on Semantic and Pattern Mining

Sentiments in material may be detected utilizing semantic methods that make use of grammatical feature learning, such as parts of speech (POS) learning. [8] This semantic technique was utilized to determine that the combination of adjectives and adverbs performs better than examining the adjectives alone. The semantic technique was utilized by Turney [74] to classify reviews. Using a syntactic parser and an emotion lexicon, Nasukawa and Yi [58] were able to accurately identify the feelings expressed in the text. Word meaning disambiguation, chunking, and the n-gram were used to classify binary polarity in Bloom, Garg, and Argamon [10]. Ohana and Tierney [59] used SentiWord Net to classify the sentiments of reviewers [59]. They used SentiWordNet as a source to generate a dataset of relevant features that counted positive and negative phrase scores to assess sentiment orientation. SentiWordNet was used by Saggion and Funk [64] to construct a collection of tools for interpreting and classifying views. Semantic orientation data was gathered from a large corpus by Hatzivassiloglou and McKeown [35] and used to automatically obtain the data. For the purpose of automatically identifying favorable and negative customer evaluations, Dave, Lawrence, and Pennock [24] created a probability-based scoring mechanism. They employed rainbow text classifiers and lexical replacements for negation handling to determine the review's category. 1

Analytical Techniques for Sentiment

A lot of work done in sentiment analysis at domain-level or for a particular genre–

Blog Level Analysis

For each blog post that was published, the authors assigned it one of three categories: objective, positive, or negative. [15] Using news and blog corpora, Ku, Liang, and Chen [47] investigated ways to extract, summarize, and monitor people's opinions. By integrating lexical knowledge with text categorization, Melville, Gryc, and Lawrence [54] accomplished sentiment analysis on blogs. Blog post opinion retrieval has been improved with the help of B He, Macdonald, J He, and Ounis [36]. Blogs written by Godbole and Srinivasan [32] were used to analyze large-scale sentiment. TREC 2006 Blog Track data was used by Zhang, Yu, and Meng [80] to evaluate an opinion mining system. A new strategy to extracting positive and negative adjectives from blogs has been presented by Dray, Plantié, Harb, Poncelet, Roche, and Trousset [26].

News Level Analysis

News reports often include the authors' own opinions on the people, places, and things they cover. A recent study by Godbole, Srinivasaiah, and Skiena [32] used large-scale sentiment analysis to examine how news sentiment differs depending on demographics, news source, and geographic region. Additionally, Balahur and others [6] conducted sentiment categorization for news and used a variety of ways to examine the applicability of various resources and approaches.

Analysis at the Review Level

Analysis of both product and movie review texts is included in review level analysis. Wiebe, Bruce, and O'Hara [75], Pang, Lee and Vaithyanathan [62], Wilson and Wiebe [76], Yu and Hatzivassiloglou [78], Hu and Liu [37] are some of the few English-language publications on review-level sentiment analysis. Domain adaptation for

large-scale sentiment categorization using a deep learning technique was done by Glorot, Bordes, and Bengio [30]. Domain adaptation for sentiment classifiers was the subject of research by Blitzer, Dredze, and Pereira [9], who looked at online evaluations for a variety of items.

Research at the Micro-Blog Scale

The Twitter microblog has been the primary focus of much of the microblog level study, which has relied on the tweets themselves as a data source. Sentiment analysis was carried out by Kouloumpis, Wilson, and Moore [43] utilizing Ed inburgh Twitter corpus2, the EMOT emoticon dataset, and a manually annotated data set from iSieve Corporation4 (ISIEVE). A Twitter corpus of text postings was used by Pak and Paroubek [60] to do sentiment analysis. They used Twitter API to get the data and divided it into three categories: good, negative, and neutral. Using remote supervision, Go, Bhayani, and Huang [31] were able to classify Twitter sentiment.

Indic Methods of Sentiment Analysis

Sentiment analysis in the Indian language was done by a small number of people. Here, we'll talk about a few important books. As a fallback technique for sentiment analysis in Hindi, Joshi and Balamurali [39] generated opinion annotated corpora for Hindi movies. They used these corpora to train the classifier, and then attempted to categorize a fresh Hindi document using this classifier. After translating the material into English, they employed a classifier trained on English movie reviews to try to categorize the translation. A subjective lexicon, called Hindi-SentiWordNet (H-SWN), was created and a majority score-based classification system was built. For the Hindi language, Piyush Arora [3] focused on creating resources such as blogs, annotated corpora, and an objective vocabulary. Mitali, Agarwal, Chouhan, Bania, and Pareek [57] used Negation and Discourse Relations to analyze Hindi review sentiment. Subjective Hindi lexicon was generated by Bakliwal, Arora, and Varma [4, 5] using a graph-based wordnet expansion approach. The basic seed vocabulary was additionally expanded via the usage of synonym and antonym relations. Dipankar Das and Bandyopadhyay [23] used Ek man's [27] six emotion classes at the phrase level to classify emotions in Bengali blog corpus. SentiWordNets for Indian languages were created by Amitava Das and Bandyopadhyay [21, 22] utilizing a variety of computational methodologies, such as WordNet, dictionary, corpus, or generative approaches. For the Bengali Language, they also focused on polarity identification at the sentence level.

Learning through doing and analyzing one's own feelings

Settles [69] discusses a wide range of active learning algorithms in the literature, including pool-based, stream-based, query synthesis, active class selection, and many more [69]. Pool- and stream-based active learning are two of the most popular active learning situations. stream-based active learning provides the learner with a continuous stream of unlabeled data items [28, 18, 16]. The learner must determine whether or not to ask for a name for a new unlabeled point on each repetition of the task. It has been our primary emphasis in our research to adopt a pool-based active learning approach [68], in which the learner is provided with both a labeled pool and an unlabeled pool at the beginning, and is then able to access a pool of (unlabeled) examples regularly. A support vector machine-based querying function has been suggested by three separate working groups in pool-based active learning [13, 44, 73]. Current pool-based active learning studies are mostly concerned with devising a rational means of picking which instances to tag. In the case of Lewis and David [49], uncertainty sampling, the classifier queries the most uncertain case. Surveys conducted by Fu, Zhu, and Li [29] and Reitmaier and Sick [63] have examined the combinations of uncertainty sampling. A probabilistic technique (PAL) that incorporates several sources of information was suggested by Kottke, Krempl, and Spiliopoulou [42]. QUIRE by Huang, Jin, and Zhou [38] is another common technique that analyzes the representativeness and informativeness of unlabeled examples by estimating the probable label assignments. Seung, Opper, and Sompolinsky's [71, 28] Query by Committee (QBC) uses the idea of maximum disagreement across distinct classifiers to choose a query sample. According to Settles and Craven's Density Weighted Uncertainty Sampling (DWUS) [70, 67], the informative examples should include both uncertain and "representative" instances of the underlying distribution. Kolar Rajagopal [41] and others employ random sampling in which the query sample is selected at random. Few scholars have worked on sentiment analysis employing an active learning approach to learning strategies and techniques. For sentiment analysis of financial (stock market) twitter streams, a stream-based active learning technique was developed and used by Smailovi, Grar, Lavra, and nidari [72]. Crowdsourcing and active learning were utilized by Brew, Greene, and Cunningham [12] to monitor online media sentiment. There has been a lot of interest in active learning algorithms for sentiment analysis in multilingual texts [11] by Boiy and Moens [11]. Active learning was used by Etin and Amasyal [2] to study Turkish sentiment analysis.

An unbalanced class distribution situation for emotion classification where the number of negative samples differs from the number of positive samples was tackled by Shoushan Li, Shengfeng Ju, Guodong Zhou, and Xiaojun Li [50].

Resource Creation and Annotation

Data Collection

In this part, we'll look at the many sources from which raw data was collected and how that data was processed, as seen in Figure 3.1. E-commerce websites where consumers may openly voice their opinions on items and social networking sites, like Twitter and Facebook, provide the majority of sentiment analysis corpora currently accessible. Although the Sentiment Analysis community has paid far less attention to the news genre, news plays a crucial role in displaying reality and has a significant impact on social behavior. Also, a lot of Telugu data may be found on news sites. For all of these reasons, we decided to create our corpus around the genre of news. Eenadu3, Kridajyothi4, and Sakshi5 were the five Telugu news websites we scraped and gathered our raw data from. More than 453 news stories were gathered, however only 321 of them were relevant to our work. A pre-processing procedure removed headers and sub-headings as well as sentences that included non-Telugu vocabulary and cleaned out some unnecessary dots from the collected data.

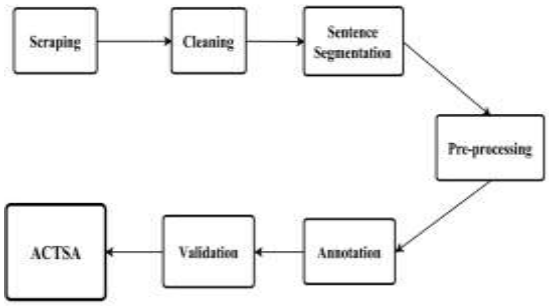


Figure 3.1 Process of building the resource

The sentences that had been gathered so far were then carefully evaluated for objectivity. Sentences that reflect no feeling, opinion, etc., are known as objective sentences. There is evidence to back up their claim. For example, the verified fact in (3.1) is an objective phrase.

Agreement Study

When we were done with the annotation, we took a test to see how dependable our system was. We performed an inter-annotator agreement analysis on the annotated sentences to test the dependability of our polarity annotation method. Annotators' judgements for each sentence are shown in agreement in Table 3.2. In order to determine Cohens' kappa, we employed

Table 3.2 Agreement for Sentences in ACTSA

Annotator 1 \ Annotator 2	Positive	Negative	Neutral	Total
Positive	1463	31	103	1597
Negative	23	1421	116	1560
Neutral	112	127	2427	2666
Total	1598	1579	2646	5823

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

Where po is the observed agreement and pe is the chance agreement. A considerable level of agreement is often defined as a number of 0.6 to 0.8. We were pleasantly surprised when the number came back at 0.87, indicating that the annotations are quite reliable.

Table 3.3 Statistics about the data

Type of data	#
News articles	321
Cleaned Sentences	11952
Objective Sentences (Removed)	4327
Uncertain Sentences (Removed)	1802
Disagreement Sentences	512
Classified	99
Removed	413
Positive sentences	1489
Negative sentences	1441
Neutral sentences	2475
Total sentences	**5410**

Corpus Statistics

In this part, we'll go through our data's statistics, starting with the raw data and working our way up to the last phrases. We scoured a number of websites for the information we needed. More than 400 news stories were gathered, however we were only interested in 321 that were related to our job. This raw data yields 11952 sentences after pre-processing. Following our subjectivity test (described in section 3.1), we eliminated 4327 objective statements, leaving us with 7812 remaining. The annotators were provided these phrases to use in their annotations, as stated in

section 3.2. Where at least one annotator considered it doubtful, 1802 sentences were eliminated. Only 512 of the total 5823 sentences were found to be in agreement, and these were sent to a third party for annotation. A total of 413 sentences were deleted after the third annotation if dispute or objectiveness prevailed. For this task, we need a corpus of 5410 sentences (1489 positive; 1441 negative; 2475 neutral) that we can annotate. Table 3.3 contains statistics on the whole corpus.

Summary

In this chapter, we took a look at how we went about gathering the data we needed for our study. Guidelines and other actions used to create the annotated corpus were explained. Manual annotations are studied by calculating the inter-annotator agreement using.

Web Content Sentiment Analysis in Telugu

As indicated in the preceding chapter, 5410 Telugu sentences were annotated by humans. It is possible to learn a lot about the quality of human-annotated data by comparing it to findings from automatic classifiers. Sentiment analysis and automated classification models were employed in this chapter to assess the dependability of our human annotated data by comparing findings from automated classification with those from our human annotated data.

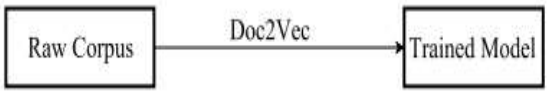
Data is obtained from other sources.

We'll go through the various kinds of data we utilized and where we got it in this part. Raw Telugu phrases are used to train the Doc2Vec model, whereas annotated Telugu sentences are used to train, test, and evaluate the classification models (refer section 4.3.2). To extract raw Telugu utterances, we used a corpus of 7,21,785 raw Telugu sentences given by the Indian Languages Corpora Initiative (ILCI). That chapter's annotated corpus (which contains 5410 annotated sentences) will be utilized in this section's final analysis.

Methodology

Step-by-step instructions for developing this method are provided here. It was our first attempt at employing several Machine Learning techniques to automatically categorize Telugu words based on their polarity, such as Logistic Regression, Support Vector Machines (SVM), Multi LayerPerceptrons (MLP) Neural Network, Decision Trees and Random Forest. A binary job of categorizing sentiment into positive and negative polarities was followed by a ternary task of categorizing sentiment into positive, negative and neutral polarity. Later parts go into great depth on the algorithm and formulation. We trained, tested, and evaluated these machine learning classifiers using a subset of our annotated data (1600 sentences / 5410 phrases). These 3810 sentences were used to verify the accuracy of our annotated data. Some 441 good and 432 negative sentences make up the total of 1600 sentences. Steps to train classifier models to automatically categorize unannotated data are outlined in the following subsections.

Figure 4.1 Training the model with raw corpus

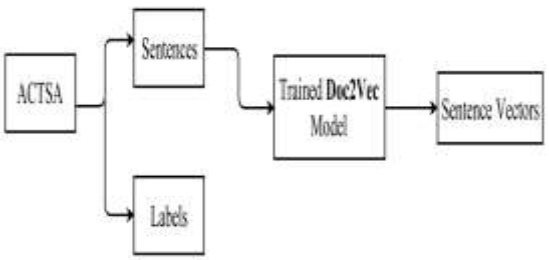


Pre-processing Phase

Step-by-step instructions for developing this method are provided here. It was our first attempt at employing several Machine Learning techniques to automatically categorize Telugu words based on their polarity, such as Logistic Regression, Support Vector Machines (SVM), Multi LayerPerceptrons (MLP) Neural Network, Decision Trees and Random Forest. A binary job of categorizing sentiment into positive and negative polarities was followed by a ternary task of categorizing sentiment into positive, negative and neutral polarity. Later parts go into great depth on the algorithm and formulation. We trained, tested, and evaluated these machine learning classifiers using a subset of our annotated data (1600 sentences / 5410 phrases). These 3810 sentences were used to verify the accuracy of our annotated data. Some 441 good and 432 negative sentences make up the total of 1600 sentences. Steps to train classifier models to automatically categorize unannotated data are outlined in the following subsections.

Training Phase

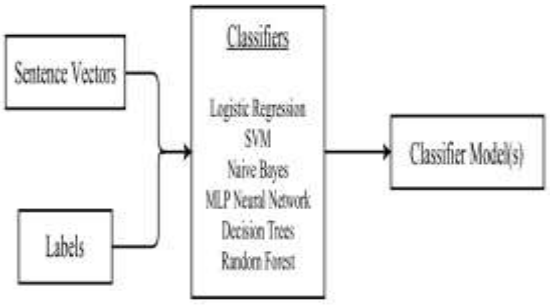
Each annotated sentence is transformed into a vector in the pre-processing step. As a result, we have a labeled vector with a matching label for each phrase. NowBuilding Sentence Vectors in Figure 4.2



The last step is to divide the data into binary or ternary categories. Section 4.3.2 explains the different Machine Learning classifiers we utilized

for this challenge. We separated the data into two sets: one for training and the other for testing, with a ratio of 4:1 between the two. The training set of sentence vectors and their corresponding labels are used to train the models of all the classifiers. As a result of this training, the model can categorize any new unannotated Telugu phrase. There is a summary of the training phase shown in figure 4.3.

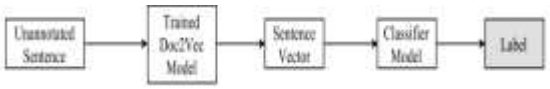
Figure 4.3 Training the classifier models



Classification

An unannotated Telugu phrase may be automatically classified using the training Doc2Vec model and the trained classifier model. Figure 4.4 illustrates the method. To get the sentence vector, the input sentence is fed into the Doc2Vec model, which has been trained to produce the sentence vector. The trained classifier model is then used to retrieve the label for the provided unannotated sentence ("Positive"/"Negative" for binary classification and "Positive/"Negative/"Neutral" for ternary classification) from this sentence vector.

Figure 4.4 Classifying an unannotated sentence



Framework

There are many Machine Learning methods utilized in this area.

Doc2Vec Tool

Sentence Vector is an unsupervised method that learns fixed-length feature representations from variable-length parts of texts, such as sentences. Each document is represented by a dense vector that is trained to predict the words in the document in the paper [48]. Text is often represented as a fixed vector in machine learning techniques. This is the most frequent fixed-length vector form of texts, and it is called a bag-of-words (BOW). These representations are employed because they are simple and accurate. We are not employing bag-of-words since this strategy has several downsides. The word order is lost, and hence multiple sentences with the same collection of words will have precisely the same representation. The word order in shorter context is taken into account when using bag-of-n-grams; nonetheless, this method suffers from the curse of increasing dimensionality and data sparsity. Sentence vectors provide a number of benefits, including the ability to learn from unlabeled data. The word order is taken into account by sentence vectors. Doc2Vec is a tool in which sentences are turned into sentence vectors. This tool assists in pre-processing and training of data.

Decision Trees

When making choices, DTs employ a tree-like model to represent the options and their consequences. It is possible to build decision trees by labeling each non-leaf node in the tree with an input characteristic. There is a class designated on each leaf on the tree. However, due of the over-fitting of the training data, decision trees provide less accurate answers for us. For each decision tree, we used a tree depth of 20.

Tests and Findings

To determine which classifier works best in binary and ternary classification, we ran tests on a variety of machine learning classifiers. The data is separated into training and testing sets in a 4:1 ratio using the 5-fold cross-validation technique. Validation is performed on the data using the test set. To ensure the validity of the findings, the tests are repeated four times (trails). All the data is scrambled and then split into a 4:1 ratio before to each experiment to ensure unpredictability in the data. Here are the findings in tables. It's clear that the binary classification problem is well-suited to Random Forest, Logistic Regression, and SVM models. The Random Forest Classifier is popular because to its more user-friendly layout and ease of comprehension. We can see that Logistic regression performs well in a ternary categorization. Four trials with five iterations each were undertaken and the findings have been compiled. In the final column of each table, we included the average of five trials for each approach.

Innovator's Query Selection Strategy: A Hybrid

We wanted to expand the number of labelled datasets accessible since there wasn't enough annotated sentiment data in Telugu. However, manually annotating large amounts of unlabeled data is very time-consuming, expensive, and resource-consuming. One potential answer to this challenge is to use active learning. Picked examples of training data are purposefully selected to decrease the annotation effort in numerous natural

language processing tasks such as sentiment analysis and text classification, among others. Active learning for sentiment analysis in Telugu is examined in this chapter. In order to improve training data accuracy while working with sparse amounts of labeled data, we devised a hybrid technique that draws on many distinct query selection strategy frameworks. We also tried to do sentiment analysis in Telugu and assign a good or negative polarity to a particular text. To get the best results, we used SVMs, XGBoosted Trees (GBT), and Gradient Boosted Trees (GBT).

The creation of a dataset

For Telugu, there is neither a huge dataset, tools, nor pre-trained models as there are for English. To build a word embedding model and to extract data and sentiment, telugu data must be preprocessed. The Wikipedia Telugu dump, accessible in Unicode format, was utilized in this project (UTF). WX notation1 is a convenient notation for implementing and experimenting with this data, which is why it was transliterated [33]. Here is an example in both UTF and WX notation of the same sentence:ACTSA in Chapter 3 is utilized as the annotated dataset, which contains about 1000 words with just positive and negative polarity. We used a UTF-WX converter to transliterate the annotated dataset in the same way as the raw dataset. The annotated data (D) of roughly 1000 sentences was used in our example. As a starting point, we used just 200 sentences of test data (DT), with 10 phrases designated as labeled data instances (DL) and 790 sentences designated as unlabeled (DU).

Generation of Word Embeddings

To construct the word embedding model, we turned to the word2vec [56] method. The following word in a phrase may be predicted with the help of word vectors [55]. Our first word embedding model was built using the Telugu raw dataset (in WX notation). t-SNE is a method for dimensionality reduction that may be used to visualize high-dimensional vectors, such as word embeddings, to verify the created word embeddings. The word embedding model was used to create a 100-dimensional feature vector for each annotated text (D).

Approaches Under Consideration

With the use of several active learning methodologies, we have developed a new methodology dubbed the Hybrid query selection approach. This method handles the challenge of picking from a pool of potential algorithms depending on how well they contribute to the learning process on a particular set of training data.

Data that is entered into the System

Data created in section 5.1.1 is sent to the system as input. The length of each sentence vector is 100 dimensions. Unlabeled data instances (DU), labeled data instances (DL), and test data were separated in the initial configuration (DT).

Sampling for Uncertainty

Uncertainty sampling is perhaps the most often utilized approach for selecting query candidates. In this method, an active learner inquires about the occasions in which it is least confident or unsure. For binary classification, the instance with a posterior probability of being positive closest to 0.5 is what uncertainty sampling looks for in probabilistic learning models. Uncertainty sampling queries the instance with the lowest prediction confidence for classifications with three or more classes. By forecasting all of the examples in the class with the greatest posterior probability upon which the instance with the least prediction is based, it determines the instance with the least amount of confidence in its predictions. Entropy may also be employed as a measure of uncertainty for the purpose of sampling [49]. For our approach, we utilized the least confident way possible.

Uncertainty Sampling via Density Weighting

DWUS is a density-weighted technique to sampling uncertainty. Querying outliers is more likely using simpler query selection algorithms like uncertainty sampling, QBC, etc., since they focus on a single instance rather than the whole input field. Due to their contentiousness or projected significance, they spend time interrogating probable outliers [70, 67]. However, the informative instance is not "typical" of the other instances in the distribution. For future failures, we may prevent these issues by using unlabeled data instances (DU). Our goal in DWUS isn't only to identify a representative example of the distribution, but to find the most informative example. In order to decrease error as rapidly as possible, we mix uncertainty with the density of the underlying data. K-means clustering was utilized to construct an initial training set of data. Pre-computed and cached densities may further decrease DWUS's query selection time, making it almost identical to uncertainty sampling.

Experiments and Results

For each classifier, we performed tests utilizing uncertainty sampling, random sampling, QUIRE, DWUS, and the Hybrid method to investigate the error rate's behavior in relation to a number of queries [19, 14]. A cross-validation procedure is used to determine the optimal set of parameters for

each of the classifiers. When adjusting parameters using cross validation, we utilized hyper-parameters derived from training and validation dataset samples as indicated in Table 5.4 since there were only a few test data instances.

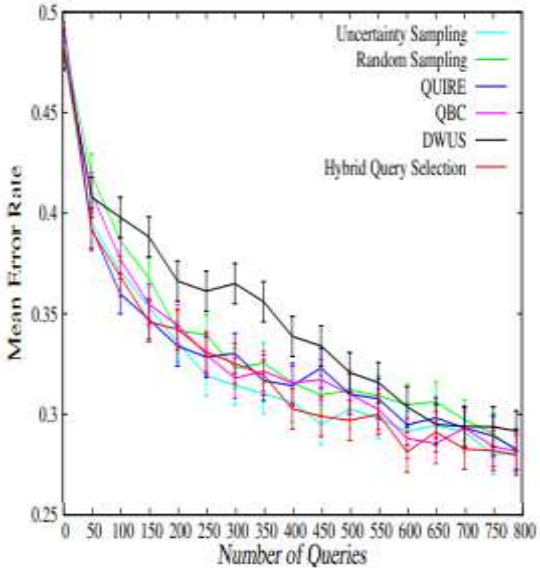
Table 5.4 Classifier parameters.

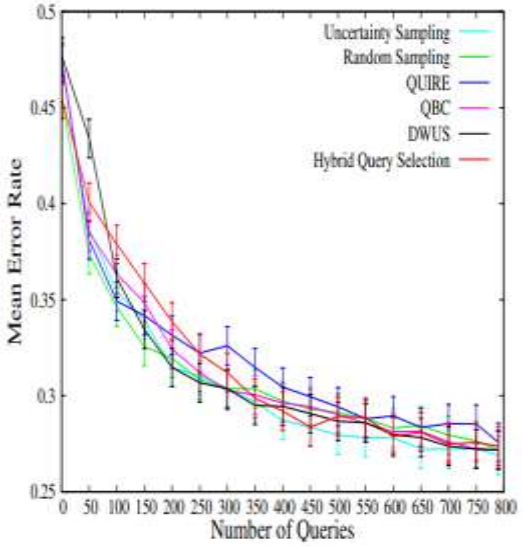
Classifier	Parameters used
XGB	N_estimators = 100, Learning rate = 1.0, Max_depth = 3
SVM	Kernel = Linear, Regularization-Parameter(C) = 10
GBT	N_estimators = 100, Learning rate = 1.0, Max_depth = 3

Mean and median error rates across 20 iterations are shown in Figures 5.6 to 5.8 and Figures 5.9 to 5.11, respectively, for a given number of queries. According to the results shown in Figures 5.6 and 5.9, the classifier XGBoost [14] uses hybrid query selection and uncertainty sampling methods, which reduces error rates as the number of queries increases. However, the same image shows that DWUS and random sampling do not reduce error rates as effectively as hybrid query selection and uncertainty sampling methods. Although the overall error rate for QUIRE and QBC is decreasing, there are too many swings in the numbers. For the SVM classifier, uncertainty sampling and DWUS functioned well and showed a continuous error decrease in Figures 5.7 and 5.10, respectively. For this reason, each new training instance of SVM increases their assessment of uncertainty. Figures 5.8 and 5.11 indicate that the GBT classifier, which we employed for Hybrid query selection, works effectively and has a low error rate. 4

Query volume vs mean error rate for each of three classifiers

Figure 5.6 XGBoost: Queries vs Mean Error Rate





GBT: Queries vs Mean Error Rate

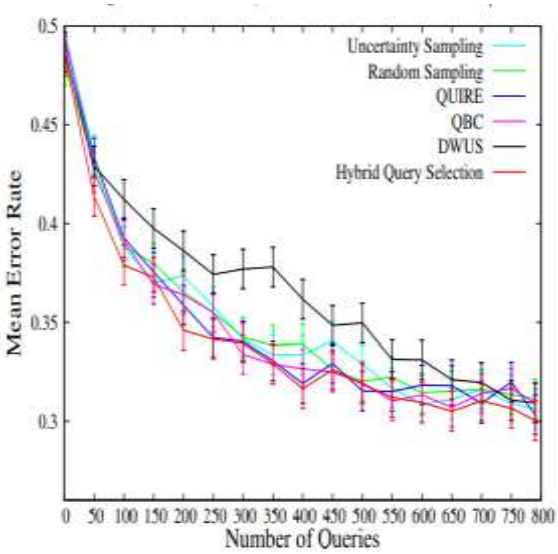
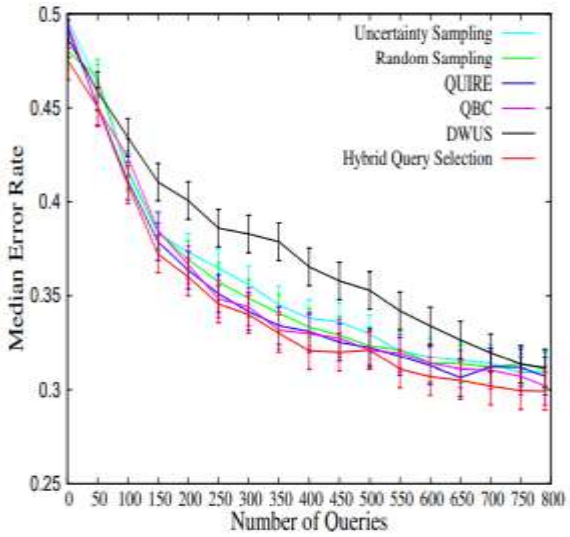


Figure 5.11 GBT: Queries vs Median Error Rate



Figures 5.6 to 5.11 show that although certain classifiers do well with each query selection approach, they do not with others. XGBoost outperforms the other classifiers in terms of learning ability and error rate reduction. The XGBoost classifier in conjunction with the Hybrid query selection strategy also performed better than other query selection approaches, as we have discovered during our testing process. XGBoost's model and the Hybrid query selection technique were thus combined for the 48-hour testing period.

Table 5.5 Minimum Average Error of each classifier by using Query Selection Strategies

Strategy	XGBoost	SVM	GBT
Uncertainty Sampling	0.310787	**0.305643**	0.349730
Random Sampling	0.332918	0.318949	0.346651
QUIRE	0.32543	0.315628	0.331627
QBC	0.327081	0.311489	0.333589
DWUS	0.343632	0.309189	0.366742
Hybrid Query Selection	**0.304801**	0.310577	**0.325397**

In Table 5.5, the average error rate for each query selection technique is shown in relation to the classifier. According to Table 5.5, the suggested hybrid query selection approach has a decreased mistake rate due to its capacity to learn the semantics of the phrases and to accurately forecast them. While SVM classifiers provide reduced error rates when used with both uncertainty and DWUS approaches, they do not perform as well when used with hybrid query selection methods. Random and QUIRE query selection techniques have a higher mistake rate than monotonically declining graphs because of the lack of assurance in picking queries. Test data samples are shown in Table 5.6 in terms of their accuracy, recall, and F-measure based on the best query selection techniques used in the training phase. We tested 200 samples and found 99 to be positive, while the other 101 were negative. Table 5.6 shows that the XGBoost classifier outperforms the other two classifiers somewhat. We propose to apply a meta-learning strategy [66] to annotate all unlabeled data since the results from all three classifiers are essentially identical.

Summary

Here, we developed a word embedding model for the Telugu language as well as analyzed the various embedding spaces. We used a little amount of labeled data to solve the challenge of annotating labels on unlabeled Telugu data. Using the advantages of active learning and a hybrid method, we tested five alternative query selection techniques and compared the error rates with each of the separate query selection strategies to address this problem. For Telugu, we used a hybrid technique and tagged a large number of previously unlabeled data points. We developed a Telugu sentiment analysis model and tested it with many different classifiers.

Conclusions

Sentiment Analysis is a hotly debated topic in the fields of data mining, web mining, and text mining, all of which rely heavily on the techniques of natural language processing. Developing better goods and services, greater knowledge of customer needs, and improved decision-making have all been aided by this newly-emerging sector. The old method of relying on word-of-mouth has been replaced by relying on online evaluations and opinions. There has been a surge in user-generated material in the Telugu language over the last few years, which has opened the door to enhanced facilities and services for the customers. For sentiment analysis in the Telugu language, one of the biggest hurdles is that there are little resources available to work with. We created and made publicly accessible an annotated corpus of Telugu sentences that meets the highest standards. Three hundred and twenty-one news pieces comprise the corpus. There are 1489 positive, 1441 negative and 2475 neutral statements carefully annotated by educated native Telugu speakers following our annotation criteria in this collection. With Cohens kappa,, the inter-annotator agreement is checked for its dependability, and the result is that there is complete agreement at 0.89. Machine Learning

techniques such as Naive Bayes, Logistic Regression, SVM (Support Vector Machines), MLP (Multi Layer Perceptron) Neural Network, Decision Trees, and Random Forest were utilized to categorize the polarity of Telugu texts. We developed models for categorizing sentiment into positive and negative polarities and positive, negative, and neutral polarities using binary and ternary classification tasks, respectively. Random Forest, Logistic Regression, and Support Vector Machines fared well for binary classification, whereas Logistic Regression did well for ternary classification. There were four trials, each with five iterations, of the assessments. The learning algorithms' performance was also assessed using the evaluation metrics Precision, Recall, and F-measure. Comparing our 51 human-annotated corpora with automatic classification results demonstrated the validity of our data. The findings were encouraging. We studied the applicability of active learning for labelingunlabeled Telugu phrases with little labelled data since the human annotation procedure is time-consuming, expensive, and resource-intensive. The multi-arm bandit issue inspired us to develop a hybrid query selection method for active learning. In each cycle, the hybrid query selection approach selects one of the following: Uncertainty Sampling, Random Sampling, Querying Informative and Rep resentative Examples (QUIRE), Density Weighted Uncertainty Sampling (DWUS), Query by Committee (QBC). We used this strategy for Telugu and labeled a large number of previously unlabeled data points. Using support vector machines (SVM), extreme gradient boosting (XGBoost), and gradient-boosted trees as classifiers, we created a sentiment analysis model for Telugu using active learning (GBT). We were able to accomplish impressive outcomes with a low rate of error. Telugu language embedding model was also constructed and tested in various locations of the embedding space.

The following are a few of the study's most significant takeaways:

1. Created and made publicly accessible a gold-standard corpus of Telugu texts annotated by hand.

2. Different Machine Learning Techniques were used to classify the polarity of Telugu sentences: Naive Bayes, Logistic Regression, SVM, MLP, Decision Trees, and Random Forest.

3. It was decided to investigate several sections of the word embedding space for the Telugu language.

4. Developed a classification model for Telugu sentiment analysis using active learning by experimenting with three different classifiers, namely, support vector machines (SVM), extreme gradient boosting (XGBoost), and gradient boosted trees (GBT) (GBT).

Future Work

Using our hybrid method, we want to double the quantity of data that has been labeled and make it accessible to the public. Transfer learning methods will be tested now that we have a significant quantity of labeled data for our experimentation. We want to separate this issue into its own domain in the future. Our goal is to apply this method universally to any language that has a limited vocabulary.

Bibliography

[1] A. Agarwal, F. Biadsy, and K. R. Mckeown. Contextual phrase-level polarity analysis using lexical affect scoring and syntactic n-grams. In Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, pages 24–32. Association for Computational Linguistics, 2009.

[2] M. F. Amasyalı et al. Active learning for turkish sentiment analysis. In Innovations in Intelligent Systems and Applications (INISTA), 2013 IEEE International Symposium on, pages 1–4. IEEE, 2013.

[3] P. Arora. Sentiment analysis for hindi language. MS by Research in Computer Science, 2013.

[4] P. Arora, A. Bakliwal, and V. Varma. Hindi subjective lexicon generation using wordnet graph traversal. International Journal of Computational Linguistics and Applications, 3(1):25–39, 2012.

[5] A. Bakliwal, P. Arora, and V. Varma. Hindi subjective lexicon: A lexical resource for hindi polarity classification. In Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC), pages 1189–1196, 2012.

[6] A. Balahur, R. Steinberger, M. Kabadjov, V. Zavarella, E. Van Der Goot, M. Halkia, B. Pouliquen, and J. Belyaeva. Sentiment analysis in the news. arXiv preprint arXiv:1309.6202, 2013.

[7] A. Balahur, R. Steinberger, E. Van Der Goot, B. Pouliquen, and M. Kabadjov. Opinion mining on newspaper quotations. In Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT'09. IEEE/WIC/ACM International Joint Conferences on, volume 3, pages 523–526. IEEE, 2009.

[8] F. Benamara, C. Cesarano, A. Picariello, D. R. Recupero, and V. S. Subrahmanian. Sentiment analysis: Adjectives and adverbs are better than adjectives alone. In ICWSM, 2007.

[9] J. Blitzer, M. Dredze, F. Pereira, et al. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In ACL, volume 7, pages 440–447, 2007.

[10] K. Bloom, N. Garg, S. Argamon, et al. Extracting appraisal expressions. In HLT-NAACL, volume 2007, pages 308–315, 2007.

[11] E. Boiy and M.-F. Moens. A machine learning approach to sentiment analysis in multilingual web texts. Information retrieval, 12(5):526–558, 2009.

[12] A. Brew, D. Greene, and P. Cunningham. Using crowdsourcing and active learning to track sentiment in online media. In ECAI, pages 145–150, 2010. 54

[13] C. Campbell, N. Cristianini, A. Smola, et al. Query learning with large margin classifiers. In ICML, pages 111–118, 2000.

[14] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 785–794. ACM, 2016.

[15] P. Chesley, B. Vincent, L. Xu, and R. K. Srihari. Using verbs and adjectives to automatically classify blog sentiment. Training, 580(263):233, 2006.

[16] W. Chu, M. Zinkevich, L. Li, A. Thomas, and B. Tseng. Unbiased online active learning in data streams. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 195–203, 2011.

[17] J. Cohen. A coefficient of agreement for nominal scales. Educational and psychological measurement, 20(1):37–46, 1960.

[18] D. Cohn, L. Atlas, and R. Ladner. Improving generalization with active learning. Mach. Learn., pages 201–221, 1994.

[19] C. Cortes and V. Vapnik. Support-vector networks. Mach. Learn., pages 273–297, 1995.

[20] A. Das and S. Bandyopadhyay. Phrase-level polarity identification for bangla. Int. J. Comput. Linguist. Appl.(IJCLA), 1(1-2):169–182, 2010.

[21] A. Das and S. Bandyopadhyay. Sentiwordnet for indian languages. Asian Federation for Natural Language Processing, China, pages 56–63, 2010.

[22] A. Das and S. Bandyopadhyay. Dr sentiment knows everything! In Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies: systems demonstrations, pages 50–55. Association for Computational Linguistics, 2011.

[23] D. Das and S. Bandyopadhyay. Labeling emotion in bengali blog corpus–a fine grained tagging at sentence level. In Proceedings of the 8th Workshop on Asian Language Resources, page 47, 2010.

[24] K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In Proceedings of the 12th international conference on World Wide Web, pages 519–528. ACM, 2003.

[25] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In Proceedings of the 23rd International Conference on Machine Learning, ICML '06, pages 233–240, 2006.

[26] G. Dray, M. Plantié, A. Harb, P. Poncelet, M. Roche, and F. Trousset. Opinion mining from blogs. International Journal of Computer Information Systems and Industrial Management Applications, 1:205–213, 2009.