PARSING SENTIMENT IN TELUGU LANGUAGE SENTENCES

G. Pratibha[1], Dr. Nagaratna Hegde[2], Ch. Abhilash Reddy[3], D. Maneesh[4]
[1]Assistant Professor, [2]Professor, [3,4]Student
[1,3,4]Department of CSE, Matrusri Engineering College ,Hyderabad ,Telangana, India.
[2]Department of CSE, Vasavi College of Engineering ,Hyderabad ,Telangana, India.

Abstract: In recent times, sentiment analysis in low resourced languages and regional languages has become emerging areas in natural language processing. Researchers are showing interest towards analysing sentiment in Indian languages such as Hindi, Telugu, Tamil, Bengali, Malayalam, etc.With the growing amount of information and availability of opinion-rich resources, it is sometimes difficult for a common man to analyse what others think about .In order to analyse the information and to see what others are thinking about the product or a service is the problem of Sentiment Analysis. Sentiment analysis or polarity labelling is an emerging field, so this needs to be accurate. In this paper, we explore various Machine Learning techniques for the classification of Telugu sentences into positive or negative polarities and also the implementation of the emoji's in the sentences are done and are classified.

Index Terms--

I. INTRODUCTION

In natural language processing(NLP),sentiment analysis is a technique that deals with analysing the sentiments, opinions ,emotions of an individual towards a product, events, news ,movies or organizations, etc. The primary task of sentiment analysis is to identify the polarity of a text in a given document. The polarity may be either positive, negative or neutral. Majority of the work in the field of sentiment classification has been done in the English Language. There has been very less research has been done for regional languages, especially Indian Languages.

Telugu is a most popular Dravidian language and are about 75 million native Telugu speakers. Telugu ranks fifteenth in the list of most-spoken languages worldwide. Currently there are a lot of web sites ,blogs etc., rich in Telugu content. Sentiment analysis can be applied to the input text in three categories namely, sentence level, document level, and aspect level. Sentence level analysis focuses on identifying sentence-wise polarity value in a given document. Document level analysis determines the polarity value based on consideration of the whole document. In aspect level analysis, it identifies the polarity of every aspect (word-wise) in a given text.In this work, we tried to classify the sentiment polarity of Telugu sentences using different Machine Learning Techniques viz., Naive Bayes, Logistic Regression, SVM (Support Vector Machines), MLP (Multilayer Perceptron) Neural Network, Decision Trees and Random Forest. We built models for two classification tasks: a binary task of classification of sentiment into positive and negative polarities and a ternary task of classification of sentiment into positive, negative and neutral polarities.

The rest of the paper is described as follows: In section 2, we discuss the previous works and related work. In section 3, we describe the datasets and classifiers used for our work. In section 4, we discuss about the methodology used in our paper which includes pre-processing, training and output. In section 5,we present the framework of our work which includes the tools and different Machine Learning techniques used in our work. In section 6, we present our experiments and discuss the results.

II. RELATED WORK

Sentiment analysis is a difficult task and a lot of research has been done in the past for English text and not been done for Telugu text with emojis. In this section we analysed some of the methodologies and approaches used for sentiment analysis and polarity classification.

To motivate more researchers towards the sentiment analysis in Indian languages, Patra et al. [15] conducted a shared task called SAIL (Sentiment Analysis in Indian Languages). In that event, many researchers have presented their method to analyse sentiment in Indian language such as Hindi, Bengali, Tamil, etc.Mukku et.al. [20] is the first reported work for sentiment analysis. They have used corpus provided by Indian Languages Corpora Initiative (ILCI) data and trained with the help of Doc2Vec model and for pre-processing, Doc2Vec tool that is used to give the representation of a sentence semantically provided by Gensim, a Python module. Machine learning techniques that are used to train the system such as SVM, regression, NB, MLP, decision tree and random forest classifiers. They have conducted experiments on binary and ternary sentiment classification.
Learning word vectors for sentiment analysis is a research work ,where Logistic Regression classifier is used as a predictor. [Maas et al., 2011] proposed a methodology which can grasp both continuous and multi-class sentiment information as well as non-sentiment annotations.

Distributed Representations of Sentences and Documents is the work by [Le and Mikolov, 2014] where they make fixed length paragraph vectors or sentence vectors which are quite useful for our work. We used the tool Doc2Vec for preprocessing the data.

III. DATASET

In this we describe the raw corpus and annotated data which are domain independent. That have been used in our experiments.

3.1 Raw Corpus

Indian Languages Corpora Initiative (ILCI) provided 7,21,785 raw Telugu sentences to corpus. These sentences were used for generating sentence vectors by training the Doc2vec model.

3.2 Annotated Data

The corpus consists of Telugu sentences each attached with a corresponding polarity tag. These sentences are used to train ,test and evaluate the classifier models. The corpus was prepared from raw data taken from the Telugu Newspapers. This newspaper raw data was first annotated by two native Telugu speakers separately. The data was then merged by a third native speaker who also validated it simultaneously. The annotation consists of three polarity tags i.e., Positive, Negative and Neutral.

We performed inter-annotator agreement using Cohen's kappa coefficient. We got the annotation consistency (k value) to be 0.92.

IV. METHODOLOGY

In this we explain the steps involved in our approach. Doc2Vec tool gives the representation of a sentence semantically with respect to dataset. This means that the sentence vector represents the meaning of the sentence. Therefore, classifying the semantic vector space according to training data can classify all the future instances of the same kind thus giving the solution to the problem of sentiment analysis.

4.1 Pre-Processing

In this we use the raw corpus data to identify the emoji's and replace the emoji with corresponding adjective from the database. We use the Doc2vec tool provided by Gensim, a python module. The sentences alone are taken from annotated data and passed through the trained Doc2Vec model. The model then returns sentence vectors for each of the sentences. Here we maintained the correspondence while converting between sentences and their tags.

4.2 Training

In this the sentence vector is obtained from the pre-processing phase and is attached to a corresponding tag. Therefore, the problem is reduced to a binary or ternary classification problem. We use machine learning Classifiers to train the sentence vectors and creates Classifier model. The models are evaluatedusing5-foldcrossvalidationwherewedividedthedata into training and testing sets in the ratio 3:1.

4.3 Output

In thiswe discuss theresultanttagforagiveninputTelugusentence. We will find the emoji from the giveninputsentence and is replaced with corresponding word from database and is converted into a sentence vector using a Doc2Vec model. This sentence vector is given to the trained classifier model which returns the output tag.

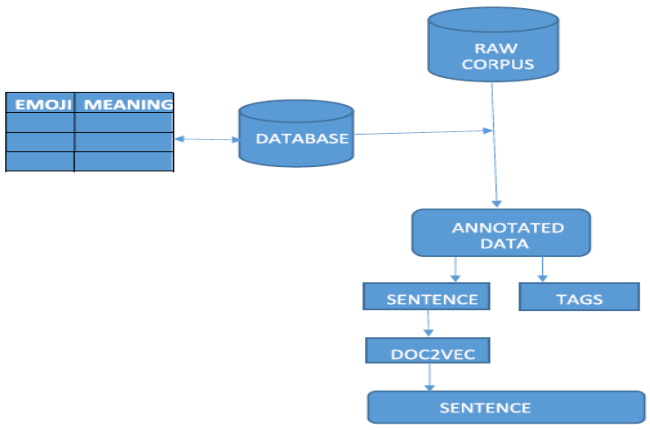

Fig 1.

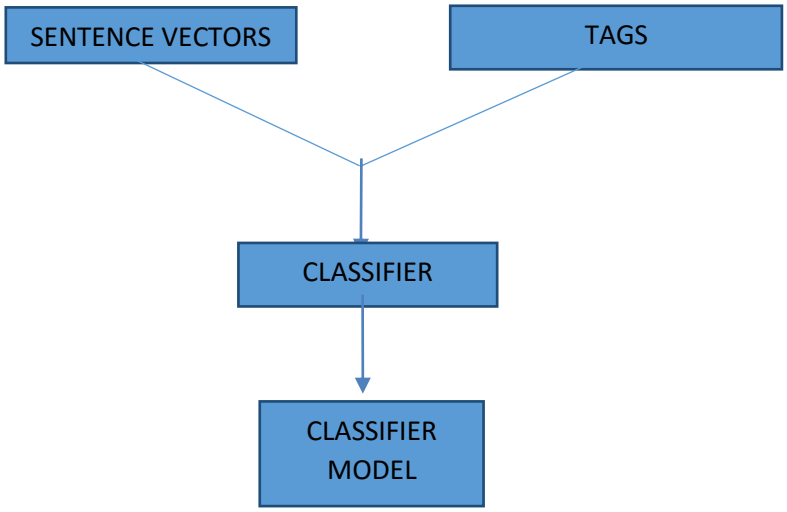

Fig 2.

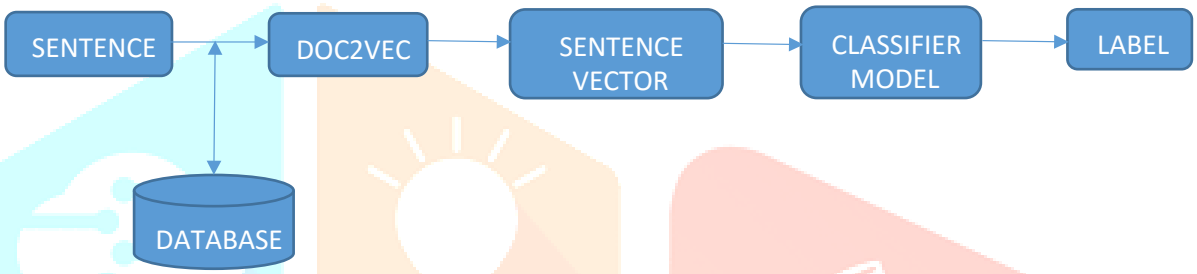

Fig 3.

V. FRAMEWORK

In this we explainthe various Machine Learning Techniques employed and the tool used.

5.1 ML Techniques

We used the following Machine learning classifiers to classify.

Naive Bayes (NB): Naive Bayes classifier uses Bayes Theorem ,which evaluates the probability of an event with respect to probability of another event which has previously occurred. This classifier works very efficiently for linearly separable and non-linearly separable problems.

Liner Regression (LR): Liner regression is one powerful classifier, which is poly class logistic model. It is used to classify an object in the predefined classes by using probability with the help of independent variables.

Support Vector Machine(SVM):SVM classifier constructs a set of hyperplanes in a high-dimensional space which separates the data into classes. SVM takes the input and for each data row it predicts the class to which this input row belongs. It is a non-probabilistic linear classifier.

Multi-Layer Perceptron :A multilayer perceptron (MLP) is a feed-forward neural network which maps input data sets to the appropriate set of outputs.Feed-forward means the data flows only in onedirection,inourcasefrominputtooutputi.e.,inforward direction.Generally neural network has three layers Input Layer, Hidden Layer and Output Layer. MLP consists of multiple layers of nodes in a directed graph, each layer is fully connected to the next layer

Decision Trees (DT): Decision tree (DT) is a tool that uses a tree like model for the decisions and likely outcomes. A decision tree is a tree in which each non leaf node acts as input feature which leads to leaf node which is label for sentence. Each leaf of the tree is labelled with a class.

Random Forest(RF): Random Forest (RF) is a group of Decision Trees. Random Forests construct multiple decision trees and considers each of their scores for giving the final output. Random Forestsreducesoverfittingasmultipledecisiontrees are involved. Decision Trees tend to overfit on a given data and hence they will give good results for training data but bad on testing data.

5.2 DOC2VEC TOOL: Doc2Vec is a tool in which sentences are converted into sentence vectors. Thistoolhelps in pre-processing and training of data. Doc2vec is an extension of word2vec that learns to correlate labels and words, rather than words with other words.

VI. RESULTS

Results are taken from five iterations from Itr0 to Itr4 by changing ratio of division of training and testing data and average is taken from all.We can increse the accuracy by increasing the training data set size shown in Table 1.

Accuracy = CS/N*100, where CS is total number of statements correctly classified and N is total number of statements.

	Itr 0	Itr 1	Itr 2	Itr 3	Itr 4	Avg
NBC	63.55	65.42	64.96	63.91	65.20	64.60
LRC	66.94	68.16	65.25	68.33	67.58	67.25
SVM	68.94	68.35	65.60	67.76	66.90	67.51
MLP	65.71	63.53	62.95	63.48	68.36	63.81
DTC	54.47	56.30	55.26	54.81	53.56	54.88
RFC	65.94	67.76	69.06	66.01	70.21	67.79

Table 1

VII. CONCLUSION

Considering the fact that we got good results and our work being first attempt on the hybrid sentences . This approach produces a more focused and accurate sentiment summary of a given Telugu hybrid sentences which are useful for the users. This approach is not restricted by any domain. However, small changes in the pre-processing would be sufficient to use this algorithmic formulation in different languages.

VIII. FUTURE WORK

• To build a dictionary for emoji based sentiment which are frequently used for positive and negative opinion and to construct a lexicon-based system .

• To test the tool for irony detection.

REFERENCE

[1] G.Pratibha, Asst.Professor, Matrusri engineering college, “An Hybrid Approach in Classification of Telugu Sentences”.
[2] Radhika Mamidi**,** LTRC, IIIT Hyderabad,”Enhanced Sentiment Classification of Telugu Text using ML Techniques”.
[3] “Scikit-learn: A machine learning in python”,http://www.Scikit-learn.org.
[4] Santosh Kumar Bharti, NIT Rourkela, “Sentiment Analysis using Telugu SentiWordNet”
[5] Chekuri Rama Rao, “Telugu Vakyam”, Andhra Pradesh Sahithya Academy, 1975.
[6] Bh.Krishna Murthy and J.P.L.Gwynn, A Grammar of Modern Telugu,. Oxford University press, 1985.
[7] Quoc V. Le and Tomas Mikolov, “Distributed Representation of Sentences and Documents.”, arXiv preprints, arXiv,1405.4053, 2014.
[8] Martin Anthony and Peter L. Bhartlett, “Neural Network Learning: Theoretical Foundations”, Cambridge University Press, 1999.
[9] Ng, Andrew Y. and Jordan, Michael I, “On Discriminative Vs. Generarative Classifiers: A Comparison of logistic regression and naïve Bayes”, Advancesin Neural Information Processing Systems 14,NIPS 2001.
[10] Zellig S Harris, Distributional Structure, Word. 10-23: 146162, 1954.
[11] Keerthi, S.S, Shavade, S. K, Bhattacharya. C & Murthy, K.R.K. “Improvements to Platt’s SMO algorithm for SVM Classifier.”, Neural Computation, vol. 13, Issue 3, March 2001, pp. 631-649.
[12] Jiang, J.J. Cornath, D.W, “Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy”, arXiv preprint, arXiv: cmp-lg /9709008, pp. 1-15,1997.