JCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.7, Special Issue.1, Jan 2019 E-ISSN: 2347-2693

Classification of Negation in Sentiment Analysis using Twitter Data

A. Dian[1*], D. Saha[2]

[1,2]Computer Science and Engineering, Jadavpur University, Kolkata, India

[*]*Corresponding Author: anik.dian32@gmail.com, Tel.: +919593727247*

Available online at: www.ijcseonline.org

Abstract— This paper provides a brief overview of a paradigm that has been used to identify, classify and cluster the negations consist in the Tweets. Usually unambiguous short text messages, collected from the famous microblogging service Twitter, are called Tweets. It has a maximum character limit of 280 characters. People usually express their standpoints or perspectives about a situation or fact through Tweets. In this collected dataset of Tweets, some negations may be overlapped or/and misclassified. So, our objective is to improve the accuracy using fine classification and increase the sharpness by reducing the overlap or/and misclassification. Here, we have used two different techniques of Sentiment Analysis, such as Lexicon Based Approach and Supervised Learning Approach to train our model. This proposed system has also analyzed Tweets and Emoticons into three categories- Positive, Negative and Neutral. In this analysis, we have used a data set of 2000 Tweets and found 88.14 percentage of accuracy.

Keywords— *sentiment, negation, polarity, emoticons*

I. INTRODUCTION

In the last few years, different forms of communication have emerged, like microblogging and text messaging. These microblogs and texts often convey large information on people’s opinions and sentiments about incidents happening around them. Sometimes these opinions are used to express positive, negative or neutral feelings about a situation or an event or a discussion. Sentiment Analysis focuses on the viewpoint of a speaker or writer regarding any subject matter or the overall circumstantial polarity or emotional response to a document, interaction or event. The study of Sentiment Analysis is not just a feature in a social media tool, but exceptionally complex if it is done properly.

One of the most crucial assignments in Sentiment Analysis is to determine the order of words affected by negation. In Natural Language Processing (NLP), negation functions as an operator, which is like quantifiers and models [1]. A main characteristic of operators is that, they have a scope which means that their meanings affect other particles in the text. It is known that negation is antithetical of affirmation. Thus we can say that, negation is the operation of making a sentence negative usually by appending negative elements within the structure. The appearance of the word negation is capable to change the polarity of the text and if it is not handled in a proper way, then it will affect the process of the Sentiment Analysis. So, we require an efficient algorithm that can properly analyze the negations hidden in those text messages. Traditional methods have used static window [2] and punctuation marks [3] to determine the scope of negation. However, these methods do not fulfill expectations due to the discrepancy in the negation scope, impotence to deal with linguistic characteristics and inappropriate word sense clarification. A general approach for negation handling in English text is: if a word x follows a negative word, then a new feature 'NOT x' is created as a tag for every remaining word until a punctuation mark is found [4]. But this method cannot identify the scope of negation properly, because it heuristically tags all the words until it finds a mark, without concerning whether it is actually a negative word or not. Another traditional method, unigram features cannot deal with the objective reality of negation, because unigram analyzes negative words with the words, those are negated as a separate word [4].

In this work, system investigates the dominance of negation in sentiment classification using the Twitter dataset and optimizes the overlapped or/and misclassified negations.

Organization of this paper is in the following manner: Section I sets up the necessity of this analysis and introduces different conventional approaches for classification of negation, Section II contains the literature survey on the classification of negation, Section III provides a piece of information about the Naïve Bayes Approach and Dictionary Based Approach, Section IV narrates the execution of the experiments and explores the results concluded by our proposed model and Section V concludes the research work with future scopes.

II. RELATED WORK

A literature review discusses the state of the art in a particular subject area. The evolution of negation in philosophy, beginning with Aristotle, is presently an area that produces an appreciable number of studies in the domain of philosophy, logic and linguistics. Research work on Negations was initiated by Frans Zwarts, in a Dutch paper called "Negatief Polaire Uitdrukkingen", in 1981 [5]. In that paper, he proposed a binary discrimination between a weak and strong entity, the weak entity being permissible in all downward necessitate contexts, whereas strong ones in a proper subset. Hence, he called the set of factors as anti-additive functions. In the year 1993, Talmy Givón proposed a functional based English grammar [6]. It is based on the grammatical subsystems repeatedly found in some simple points: Verbal inflections, auxiliaries and the grammar of tense-aspect modality and negation; articles determiners, pronouns and the grammar of referential consistency; variation of noun phrases and noun modifiers. Another work on negation handling was done by Pang et al. in 2002 [3]. They arranged the documents not by subject matter, but by sentiments to decide whether the evaluation is positive or negative. The authors used the punctuation marks to identify the scope of negation. This method upturned the polarities of those words which come between the negation word and the next punctuation mark. In the year 2009, Jia et al. worked on the effect of the individual appearance of a negation in a sentence on its polarity and bought in the notion of the effect of negating terms [1]. In the year 2010, Wieghan et al. were focused on the impact of the negation in sentiment sentences [7]. They declared that fecund negation model for Sentiment Analysis usually needs the perception of polar expression. Bojar et al., a group of Indonesian researcher proposed a research work on the resources of the lexicon for Indonesian sentiment in 2015 [8]. They did the negation handling by redesigning the method of Das and Chen. They developed a methodology for sentiment extraction from a stock message board [9]. In 2015, Wang et al. carried out a survey on negation contribution in Sentiment Analysis [10]. They concluded that the fruitful negation design for Sentiment Analysis usually needs an understanding of polar statement. During this year, Dedvar et al. investigated the scope of negation in Sentiment Analysis [11]. They concluded that established negation identification methods are deficient for the task of Sentiment Analysis and that development is to be made by utilizing the information about how opinions are expressed utterly. U. Farooq et al. did a research on negation handling at sentence level in 2017 [12]. They examined the difficulty of identifying the scope of negation while deciding the polarity of a sentence. They proposed a negation handling method based on semantic characteristics which determine the performance of various types of negation.

III. SENTIMENT ANALYSIS TECHNIQUES

A. API Based Sentiment Analysis

We have used AYLIEN API to analyze the individual Tweet i.e. to find the type of opinion (positive, negative or neutral) conveyed by a Tweet. This text analyzing API includes some distinct methods of Natural Language Processing, knowledge gathering and Machine Learning processes which give us the facility to extract the meaning and insight view of the Tweets. It goes through the following processes to extract the meaning of a Tweet: article extraction, summarization and classification using the IPTC News Code Standard, entity separation, hypothesis extraction, language recognition and Sentiment Analysis.

B. Dictionary Based Approach

The Dictionary Based Approach depends on finding keywords by consulting with a dictionary and gathers a set of opinion words. We have consulted with the six types of dictionaries for doing the counting and clustering process: a Positive word dictionary, a Negative word dictionary, a Negative contraction and Auxiliary's dictionary, an Acronym dictionary of more than 500 acronyms with their translations, an Emoticon dictionary with more than 1000 emoticons and a stop word dictionary with around 10000 stop words.

C. Naïve-Bayes Classifier

A simple probabilistic classifier based on Bayes' theorem with strong (naive) independence presumptions is called Naive Bayes classifier. Naive Bayes classifier is a highly expandable method. It is an elementary approach for creating classifiers. It requires a number of linear parameters and a number of variables (features/predictors) in its learning problem. It is a model that allocates class labels to problem specimens and characterizes as vectors of attribute values. The class labels are produced by some previously defined finite sets. Let us consider c as Hypothesis, d as Tuple and P (c|d) as the probability of c conditioned on d. Then, by Bayes rule we can write:

$$P(C=c|D=d)=\frac{P(D=d|C=c)P(C=c)}{P(D=d)} \quad (1)$$

Where P (c) is the prior probability of c i.e. the probability value that c holds is true irrespective of the Tuple value and P (d) is probability of d for a given Hypothesis.

In our case, a Tweet d can be amount to a vector of K attributes, such as d = (w, w..., w). Computation of P (d|c) is not trivial and that is why Naive Bayes introduces the assumption that all of the feature values w_j are independent at the given category label c. That is, for $i \neq j$, w_i and w_j are conditionally independent at the given category label c. So, the Bayes rule can be rewritten as:

$$P(c|d) = \frac{P(c) \times \prod_{j=1}^{k} p(w(j)|c)}{P(d)} \quad (2)$$

IV. PROPOSED WORK

Twitter data collection, pre-processing and cleaning:
We have collected around 100000 Tweets in the month of February (From 18th Feb to 25th Feb of 2018). These collected Tweets are based on live events happening throughout the world. We have made our own application by using the Twitter API and interacted with Twitter services through the REST API. The API provides features to access different types of data and we can easily collect Tweets to store them in the system. By default, the data was in JSON format and we have changed it to text format for easy access. The collected Twitter data were unstructured and noisy in nature, so to get more benefit or better understanding, we have performed the cleaning process through the following steps: a) HTML character escaping, b) removal of unnecessary punctuations, c) removal of other words (retweet count, Hashtags, username etc.) and d) removal of junk Tweets.

A. *API Based Sentiment Analysis:*

We have selected 2000 Tweets in the English language randomly from our previously collected dataset and analyzed those Tweets with the help of AYLIEN API whose output is given in Table 1. This has given us an understanding of the whole data set. In another word, this survey has helped us to know the attitude or the point of view of the users or the sense of the Tweets.

Table 1. Example of Sentiment Analysis through API

Tweets	Sentiment
I have to go baby girl, I will send you more cc if you want me too, goodnight beautiful	Positive
I hate leaving my boo bear :((even doe I see him this Friday LOL	Negative
Hey, can you please be here for me?	Neutral
i miss it :(i miss the people :(i miss the place	Negative
phone Screen is Being Weird for Some Fucking Reason	Negative
I miss my puppy :-(	Negative
I respect that you actually explained the viewpoint. thank you.	Positive
A Florida woman recounts how a cockroach crawled into her ear and it took nine days to get it out.	Neutral
Universal Studies in a couple weeks, Post Malone in June, LA in July, plus a beach trip with my aunts...	Neutral

From the above analysis, we have counted the number of positive Tweets, the number of negative Tweets and the number of neutral Tweets. The result of this counting procedure is given below in Table 2.

Table 2.Statistical analysis

Description	Count
Total Number of Tweets	2000
Number of positive Tweets	223
Number of negative Tweets	595
Number of neutral Tweets	1182

From Table 2, we have calculated the percentage of negative Tweets is 33.13% i.e. one-third of the whole Twitter data. These negative Tweets contain negative words. Some positive and neutral Tweets may also contain some negative words. So, we have to find those negative words and impact of those words on Tweets. First, we have done another counting process to find the total number of positive, negative and neutral words and then, a clustering process to differentiate among separate groups with useful features using these 2000 Tweets.

B. *Dictionary Based Approach*

1) Counting process:

A python script has performed tokenization of text using “nltk” and “word-tokenizer” to gather keywords. Also, it has performed the removal of stop words by using stop word dictionary. We have tried to match those keywords with the previously downloaded dictionaries (Positive, Negative, Acronym and Negative contraction, and Auxiliary’s dictionary), and if it is not present in any of these dictionaries then synonyms of that keyword were found. Then, again we have tried to match those synonyms with the dictionaries and if a match is found, we have stored those keywords in our bag of words and put them into a separate file. Then we have done a process that counts the number of positive, negative and neutral words and the results are given in Table 3.

Table 3. Counting of words

Description	Results
Total number of Tweets collected	2000
Total number of words	28935
Total number of stop words	24689
Total number of positive words	365
Total number of negative words	809
Total number of negative contractions	287
Total number of acronym	632
Neutral words(remaining words)	2153

From Table 3, this is clear that the percentage of negations (negative contractions and negative words) used in the Twitter data set is 3.78% whereas the percentage of positive words used in Twitter data is 1.26%. So, use of negative words is higher as compared to the positive words. From the above statistical survey, we can conclude that negations have more impact in Sentiment Analysis.

We have also done a counting process on Emoticons. Emoticons are actually ASCII art. These are also called Smileys or Emojis. In recent years Emoticons are very frequently used to express the feelings in the social media like Twitter. We have found a total 7458 number of Emoticons, covering 51 different types from 100000 Tweets. Here, in Table 4 we have selected only those Emoticons that have a high frequency (more than 0.8%) which resulting in 24 most commonly used Emoticons.

Table 4. Counting of Emoticons

<table>
<tr><th>Emoticons</th><th>Counting</th><th>Emoticons</th><th>Counting</th></tr>
<tr><td>:)</td><td>3460(46.4%)</td><td>‘:]</td><td>111(1.5%)</td></tr>
<tr><td>:(</td><td>1066(14.3%)</td><td>-_-</td><td>100(1.46%)</td></tr>
<tr><td>:D</td><td>857(11.5%)</td><td>=-s</td><td>84(1.13%)</td></tr>
<tr><td>;)</td><td>462(6.2%)</td><td>;(</td><td>82(1.1%)</td></tr>
<tr><td>:-=</td><td>305(4.1%)</td><td>:,’[</td><td>74(1.0%)</td></tr>
<tr><td>:@</td><td>290(3.9%)</td><td>:((</td><td>74(1.0%)</td></tr>
<tr><td>xD</td><td>238(3.2%)</td><td>:-/</td><td>70(0.97%)</td></tr>
<tr><td>:P</td><td>216(2.9%)</td><td>=]</td><td>67(0.9%)</td></tr>
<tr><td>(:</td><td>137(1.84%)</td><td><3=></td><td>67(0.9%)</td></tr>
<tr><td>:|</td><td>134(1.8%)</td><td>:”D</td><td>60(0.82%)</td></tr>
<tr><td>:></td><td>126(1.7%)</td><td>=D</td><td>60(0.82%)</td></tr>
<tr><td>]:D</td><td>124(1.69%)</td><td>DX</td><td>59(0.8%)</td></tr>
</table>

2) Clustering process:

A cluster means a grouping of similar objects. Clustering is an unsupervised machine learning method that attempts to find the natural grouping. It tries to figure out the group or label of the unlabeled data. We have a set of unlabeled data that contains a list of positive words, a list of negative words and a list of neutral words. We have tried to find out the cluster of those unlabeled data by using WordNet(Version 2.1) and Text-Blob. WordNet is used to inspect similarity among the words and also to compute the syntactic category of the verb, adjective, adverb and noun. Text-Blob is used to find the subjectivity and polarity of a sentence. It is a Python library for processing textual data and it focuses to give an access to routine text processing operations through a common interface. We have examined the words, how did they effect on the sentence and inspected, if these are capable to change the entire polarity of a Tweet or not. Also, we have attempted to find whether the word is adverb, adjective, verb or noun and if the word is deleted from the sentence then does it change the polarity of that sentence or not. So, we have selected a word from our bag of words. If it is a new word, we have passed it through WordNet to find the sense of the word and along with this, we have computed the polarity and subjectivity of the sentence containing that word through Text-Blob. In this way, we have attempted to find similar types of words having almost the same sense, polarity and subjectivity, and put them into the same cluster. As a result, we have got 7 clusters which are given in Table 5.

Besides this, we have done another clustering on Emoticons. Since Emoticons are strong, understandable and common signals of communication on social media, nowadays these have been used widely. Mainly because if a user reacts by Emoticons it looks like a real human face, even these are okay with business settings. Actually, it helps the user to communicate faster. But, these Emoticons are capable to change the polarity of a whole text. So, we have tried to know the actual meaning conveyed by the Emoticons with help of the words that belong to the same cluster. To do the clustering of Emoticons, we have taken a help from Text-Blob to find polarity and subjectivity of a given text consisting Emoticons and Emoticons dictionary to find the meaning of Emoticons. Clustering of Emoticons is given in Table 6.

Table 5. Clustering of words

Cluster	Words
A	amazed, beautiful, darling, enjoy, good, happy, sweetheart
B	advantage, bliss, calm, cheerful, cool, friendly, kindly, quiet
C	bff, lol, a little bit, less positive,gn, unbiased, dream, move-on
D	btw,b4,cre8,da,home,hello, selfiii, smart phone , stuck ,u
E	aren’t, can't, couldn't, haven't, hadn't, shouldn't, wasn't, won't
F	angry, awful, bad, bitch, bloody, break-up, cry, damn, deadly, fuck, kills, suck, scandal, stupid
G	bullshit, careless, cheat, disappointments, lie, mistakes, odd, painful, sack, vulnerable

In Table 5, cluster A, B, and C contain positive words, cluster D contains neutral words, E, F and G contain negative words. During this analysis, we have found various types of negations like-Syntactic Negations (e.g.-no, rather, couldn’t, didn’t, wasn’t, not, nowhere), Diminishes (e.g.-little, rarely, scarcely, hardly), Morphological (e.g.-dislike, irregular, immature, shameless), Auxiliary Negation (e.g.- This should not happen, This is a bad idea to go for a picnic), Noun Phrase Negation (e.g.-No credit cards are accepted for the next few days, Not many people came to attain the meeting last week), Adverb Negation (e.g.-She never apologize for her wrong behavior), Negate Adjective Phrases (e.g.-Her boss was not friendly) and Double Negation (e.g. - I don’t have anything).

We have got some common spelling mistakes done by the users, like- lose (correct-loose), alot (correct-a lot). There is confusion between the use of affect and effect, to and too, there and they’re, your and you’re. Sometimes users are not fully aware of its connotation (the feelings associated with the word), which leads to use of inappropriate synonyms (e.g.-confusion between the use of courageous or confident, conceited or vintage, old or decrepit). These types of mistakes sometimes change the polarity of the Tweets. We have found many Acronym words, e.g.-gn (means good night), bff (means best friend forever), lol (means lots of laugh), b4 (means before), gr8 (means great), omg (means oh my god), btw (means by the way) etc. Acronyms are shortened forms of words or phrases that help to communicate faster. That is the reason; the use of Acronym is very common among the users. We have noticed that negative contractions (e.g.-can't-cannot, hadn't-had not, couldn't-could not, wouldn't- would not, shouldn't-should not etc.) are used very frequently.

Table 6. Clustering of Emoticons

Cluster	**Emoticons**
A	:D :) xD
B	;) =D :"D :> : P <3=>
C	(: =]
D	:| -_-
E	:(:((:@ :,'[
F	]:D :-= ':]
G	=-S DX

In Table 6, clusters A, B, C contains mainly positive Emoticons, cluster D consists of neutral Emoticons whereas E, F, G contain negative Emoticons. The percentage of negative Emoticons (evaluated using Table 3) is 26.79%.

C. Naïve-Bayes Classifier

We already have labeled data (from Table 5). Now, we have used Naive-Bayes Classifier to train our labeled data set for future prediction. We have used the Waikato Environment for Knowledge Analysis (WEKA), a machine learning tool to do Naive-Bayes Classification. Initially, we have started with creating a CSV file that is filled up by 1096 negations, labeled with their cluster (E, F, and G) and corresponding subjectivity and polarity (of the text containing that negation, by using Text-Blob). CSV file is converted into an attribute-relation file format (ARFF) to make the dataset compatible for validation. Then we have loaded the test file in WEKA and executed the model on the test set. Here, the Naive Bayes classification is executed with test mode: 10-fold cross-validation and numeric is taken up to 4 decimal. The result is given in Table 7.

Table 7. Naïve Bayes Classification on negations

Correctly Classified Instances	889
Incorrectly Classified Instances	207
F-measure	0.772
Recall	0.794
Precision	0.751
Accuracy	81.11%
Total number of instances	1096

We have found that 207 words were incorrectly classified that means those words are not actually negations; they may be overlapped with positive or neutral words. Our goal is to find an actual number of negations i.e. increase the intensity of the classification. So, we have added those 207 words into the positive words as well as into the neutral words. Then, we have examined two cases: one with 572 positive words (207 incorrectly classified words + 365 positive words) and another with 2360 neutral words (207 incorrectly classified words + 2153 neutral words), labeled with their corresponding cluster, polarity and subjectivity. The Naive-Bayes classification has given us two results which are shown in Table 8 and Table 9.

Table 8. Naïve Bayes Classification on positive words

Correctly Classified Instances	475
Incorrectly Classified Instances	97
F-measure	0.793
Recall	0.815
Precision	0.773
Accuracy	83.04%
Total number of instances	572

Table 9. Naïve Bayes Classification on neutral words

Correctly Classified Instances	1875
Incorrectly Classified Instances	485
F-measure	0.751
Recall	0.774
Precision	0.732
Accuracy	79.44%
Total number of instances	2360

From the above two cases, we have studied that total (97 + 485) = 582 words are misclassified. So, we have combined those 582 words with 1096 negations (1678 instances) again and inspected the result of Naive-Bayes which is in Table 10.

Table 10. Final result Naïve Bayes Classification

Correctly Classified Instances	1479
Incorrectly Classified Instances	199
F-measure	0.827
Recall	0.847
Precision	0.809
Accuracy	88.14%
Total number of instances	1678

From the above Table 10, we have analyzed that total 1479 numbers of negations are present out of 28935 numbers of words. So, the actual percentage of negations is 5.11%.

V. CONCLUSION and Future Scope

We have used Naive–Bayes classifier to predict the class of individual negation. The predicted class obtained from the Naive Bayes is compared with the predicted class obtained from the Dictionary Based Approach and gives us the accuracy of 88.14%. Our performance may deflect at some point, mainly due to the occurrence of spelling mistakes, some words are used as both negative and positive sense (overlapping), use of Acronym words and improper choice of words. The results of this work present a limited view of the phenomenon. More investigation needs to be done in order to accept or reject these findings, using larger samples and real-time analysis. It will help to get more accurate results. Right now, we are focusing on calculating a single probability for each word. Instead of this, we can calculate multiple probabilities for each according to the Part of Speech. We can also make an effort to incorporate POS information within our working Machine Learning models in the future.

REFERENCES

[1] L. Jia, C. Yu, W. Meng, "*The effect of negation on sentiment analysis and retrieval Effectiveness*", In the Proceeding of the 18th ACM conference (CIKM '09), Hong Kong, China, pp.1827-1830, 2009.

[2] A. Hogenboom, P. V. Iterson, B. Heerschop, F. Frasincar, U. Kaymak, "*Determining negation scope and strength in sentiment analysis*", In the Proceedings of 2011 IEEE International Conference on Systems, Man, and Cybernetics, Alaska, USA, pp. 2589-2594, 2011.

[3] B. Pang, L. Lee, S. Vaithyanathan, "*Thumbs Up?Sentiment Classification Using Machine Learning Techniques*", In the Proceedings of the 7th conference on Empirical methods in Natural Language Processing (EMNLP '02), Philadelphia, USA, pp.79-86, 2002.

[4] R. Amalia, M. A. Bijaksana, D. Darmantoro, "*Negation handling in sentiment classification using rule-based adapted from Indonesian language syntactic for Indonesian text in Twitter*", Journal of Physics: Conference Series, Vol.971, Issue.1, 2018.

[5] F. Zwarts, "*Negatief polaire uitdrukkingen I*", Journal of GLOT, Vol.4, pp.35-132, 2002.

[6] T. Givón, "*English grammar: A function-based introduction*", John Benjamins Publishing Company, Amsterdam, Netherlands, pp.1-318, 1993.

[7] M. Wiegand, A. Balahur, B. Roth, D. Klakow and A. Montoyo, "*A Survey on the Role of Negation in Sentiment Analysis*", In the Proceedings of the Workshop on Negation and Speculation in Natural Language Processing, Uppsala, Sweden, pp.60-68, 2010.

[8] O. Bojar, Franky, K. Veselovská, "*Resources for Indonesian Sentiment Analysis*", Journal of The Prague Bulletin of Mathematical Linguistics, Vol.103, Issue.1, pp.21-41, 2015.

[9] S. R. Das, M. Y. Chen, "*Yahoo! for Amazon: Sentiment Extraction from Small Talk on the Web*", Journal of Management Science, INFORMS, pp.1375-1388, Vol.53, No.9, 2007.

[10] H. Wang, A. J. Castanon, "*Sentiment analysis via emoticons using twitter data*", In the Proceedings of 2015 IEEE International Conference on Big Data, Santa Clara, CA, USA, pp.2404-2408, 2015.

[11] M. Dadvar, C. Hauff, F. D. Jong, "*Scope of Negation Detection in Sentiment Analysis*", In the Proceedings of the 11th Dutch-Belgian Information Retrieval Workshop (DIR 2011), Amsterdam, Netherlands, pp.16-20, 2011.

[12] U. Farooq, H. Mansoor, A. Nongaillard, Y. Ouzrout, M. A. Qadir, "*Negation Handling in Sentiment Analysis at Sentence Level*", Journal of Computers, Vol.12, Issue.5, pp.470-478, 2017.

Authors Profile

Mr. Anik Dian is a student of Jadavpur University in Computer Science and Engineering department. He is pursuing his MCA degree from Jadavpur University. He had completed Bachelor of Science in Mathematics from Presidency University in 2015.

Dr. Diganta Saha is an eminent Professor of Department of Computer Science and Engineering at Jadavpur University, Kolkata. His field of specializations is Machine Translation, Language Engineering, Mobile Database Management, Text Processing, Text Classification, and Text Data Mining. He has 2 Book chapters, 15 Journals, and 54 Conferences publications. He has carried out 7 Projects under UGC and DST. Home Page: http://www.jaduniv.edu.in/profile.php?uid=660

