A Comprehensive Study of Classification Techniques for Sarcasm Detection on Textual Data

Anandkumar D. Dave,
Department of Information Technology,
Dharmsinh Desai University,
Nadiad, India
a_dave1986@yahoo.com

Prof. Nikita P. Desai,
Department of Information Technology,
Dharmsinh Desai University,
Nadiad, India
npd_ddit@yahoo.com

Abstract*— During the last decade majority of research has been carried out in the area of sentiment Analysis of textual data available on the web. Sentiment Analysis has its challenges, and one of them is Sarcasm. Classification of sarcastic sentences is a difficult task due to representation variations in the textual form sentences. This can affect many Natural Language Processing based applications. Sarcasm is the kind of representation to convey the different sentiment than presented. In our study we have tried to identify different supervised classification techniques mainly used for sarcasm detection and their features .Also we have analyzed results of the classification techniques, on textual data available in various languages on review related sites, social media sites and micro-blogging sites. Furthermore, for each method studied, our paper presents the analysis of data set generation and feature selection process used thereof. We also carried out preliminary experiment to detect sarcastic sentences in "Hindi" language. We trained SVM classifier with 10X validation with simple Bag-Of-Words as features and TF-IDF as frequency measure of the feature. We found that this simple model based on "bag-of-words" feature accurately classified 50% of sarcastic sentences. Thus, primary experiment has revealed the fact that simple Bag-of-Words are not sufficient for sarcasm detection.

Keywords*— Sentiment Analysis, Opinion Mining, Sarcasm Detection,Natural Language Processing,Text Processing, Machine Learning.

I. Introduction

Sarcasm detection is one of the challenges in Sentiment Analysis (SA). Today, many social media sites like Facebook and Twitter have allowed users to express their emotion in their language with their label to their messages. Apart from that these sites also provides application support to handheld devices like mobiles and tablets. These sources of user's emotion have the feat in area of Sentiment Analysis and Opinion Mining (OM). SA and OM have quite differentness [25]. Opinion Mining extracts and analyze people's opinion about an entity while Sentiment Analysis identifies the sentiment expressed in a text then analyze it. Therefore, the target of SA is to find opinions, identify the sentiments they express, and then classify their polarity [24].

Sarcasm is the part of British culture. Sarcasm is characterized as the ironic or satirical wit that is intended to insult, mock, or amuse [19]. Sarcasm transforms the polarity of an apparently positive or negative utterance into its opposite [16]. Sarcasm is an elegant way of the conveying message in implicit manner that makes it hard to detect. Difficulty in discernibility for sarcastic sentences may create problems for many Natural Language Processing based system such as online review summarization systems, dialogue systems or brand monitoring systems due to the failure of state of the art sentiment analysis systems to detect sarcastic comments [18]. SA can be performed at three levels: document-level, sentence-level, and aspect-level. However, sentences can be considered as a short document, which removes fundamental difference between document-level and sentence-level SA.

In our paper, we have focused on detection of sarcastic sentences presented in the textual form like "I love being ignored" using machine learning techniques. The central issue in sarcasm detection is the data set availability and their analysis. The principle wellsprings of information are from the product reviews. These studies are critical to the business holders as they can take business choices as indicated by the examination aftereffects of user's assessments about their items. Review related sites provide these data. Sarcasm detection can be applied to product reviews but can also be applied to stock markets, news articles, or political debates. In most of the cases, social media sites and micro-blogging sites provide an excellent platform for users to express their sentiment and opinion freely and which, can be a good source of information. In past few years in the field of SA remarkable work has been done but sarcasm detection in textual sentences is not up to the mark.

In this paper, we discuss different machine learning techniques for detecting sarcastic sentences in various languages, their features, Measures, dataset generation and their scope. In this study following work is carried out by us and explained in following sections:

- Identify the textual form of sarcasm.
- Study different classification techniques possible for sarcasm detection.
- Comparison of performance of these methods for sarcasm detection.
- Experiment of Sarcasm detection in Hindi Sentences and results.

This paper might be very helpful to Naïve researchers in this field.

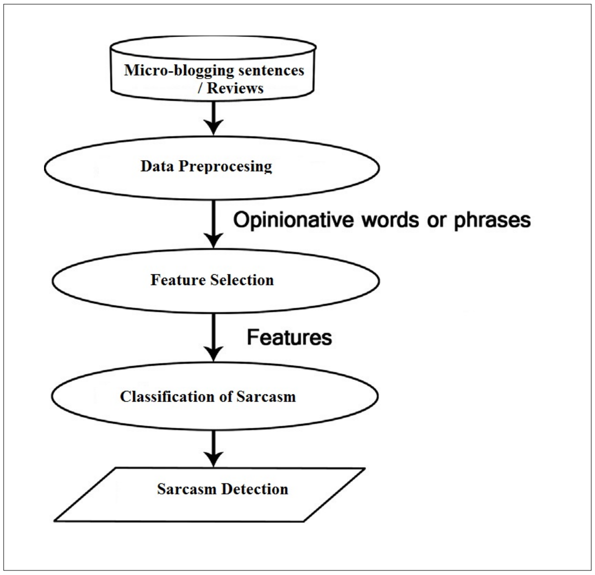
978-1-4673-9939-5/16/$31.00 ©2016 IEEE

II. Challenges and Types Of Sarcasm detection Methods

In recent years SA is a trending research area. However, sarcasm detection in textual sentences has also gained the interest of researchers and some work also has been carried out in this area. Sarcasm is defined as: "Sarcasm is a form of ironic speech commonly used to convey implicit criticism with a particular victim as its target" (McDonald, 1999, 486-87). Sarcastic sentences express the negative opinion about a target using positive words.

- *I am getting blazing internet speed here!!*

Sarcasm detection plays a significant role for refinement of the Sentiment Analysis Systems. The process of sarcasm detection using classification techniques are illustrated in Fig. 1.



Micro-blogging sentences / Reviews
Data Preprocesing
Opinionative words or phrases
Feature Selection
Features
Classification of Sarcasm
Sarcasm Detection

Fig. 1. Process of sarcasm detection

A. Sarcasm Detection: Challenges

Classification of sarcastic sentences is a difficult task. The preliminary dataset for such problem is sentences collected from online review sites, social media sites, and micro-blogging sites. These sentences are short in nature, ambiguous, unstructured i.e. not following correct grammar rules of language in which they have written. Not only the message but sentences may contain URLs, User Name, User defined label, etc. Over and above these issues, there exist particular challenges in sarcasm detection, some of which are mentioned below [25]:

- In spoken statement sarcasm can be identified with particular notation or facial expression but for written text no such clues can be found which make it is difficult to detect.
- In sarcastic statement, positive words are used to convey negative opinion.
- In some case of sarcasm detection world knowledge is required e.g. *"Yet, another mind blowing performance of Indian team in Srilanka."*
- Sometimes sarcasm uses the hyperbole. Hyperbole is the use of exaggeration i.e. use of words belonging to superlative degree e.g. *"Extraordinary performance in exam!!"*

B. Sarcasm Detection: Approaches

Various classification techniques based on their approach for Sarcasm detection in textual data are listed below:

- Lexical Analysis – #Sarcasm, use of popular sarcastic phrases.
- Likes and Dislikes Prediction – Fuzzy techniques.
- Fact Negation – tweets contradicting a fact.
- Temporal Knowledge Extraction – tweets contradicting facts about event.

III. Related Work

In the area of psychologists, behavioural scientists and linguists sarcasm is studied well [20]. But automatic detection of sarcasm is challenging task in case of text mining [16] and has been addressed by few researchers. Pang and Lee, 2008 has provided comprehensive study in field of opinion mining. In spoken form, Tepperman et al. (2006) has identified the sarcasm based on 'yeah-right' pronunciation. Kreuz and Caucci (2007) studied the influence of different lexical factors like interjections and punctuation symbols in recognizing sarcasm in written form. Filatova (2012) presented a detailed description of sarcasm corpus creation with sarcasm annotations of Amazon product reviews.

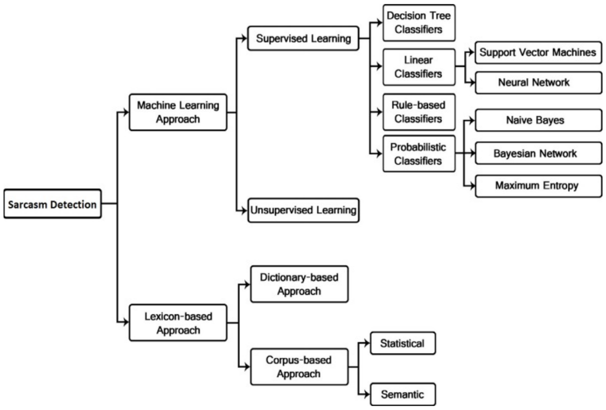


Fig. 2. Sarcasm Detection Techniques

Tsur et al. (2010) have proposed a framework for automatic detection of sarcastic and non-sarcastic sentences in Amazon reviews. Their framework exploits syntactic and pattern based feature in sarcastic sentence of Amazon product review. Devidov et al. (2010) have used sarcastic Twitter message and Amazon product review to train the classifier using syntactic and pattern based feature [18]. They have studied the reliability of sarcasm hash-tag as the golden standard for evolution but found that hash-tags are noisy in nature. Work done by [16], [17], [18], [19],[20] and [21] have used traditional supervised classification techniques like Support Vector Machine (SVM), Conditional Random Field (CRF), and Naïve Bayes along with some Lexical-based approach and concise analysis is represented in Table 1.Sarcasm detection techniques are delineated as in Fig. 2 [24].

IV. Classification of Sarcasm: The Process

Classification process of sarcastic sentence consists of several steps.

A. Dataset Generation and Data Pre-processing

Today, the enormous amount of annotated corpora is available for SA but for sarcasm detection no gold standard dataset is available which is biggest challenge in sarcasm detection. Researchers have to create this annotated corpus by human intervention. Davidov et al. (2010) have produced such corpus for their research, and raw data was collected from Amazon product reviews and Tweets. Data obtained from such online platform are unstructured and does not follow grammar rules. So, Data Pre-processing is required to remove the noise present in the data set. Noise could be a user defined label, spelling mistakes, slang words, etc.

B. Feature extraction for Sarcasm Detection

In the classification of Sarcasm on textual data, feature extraction and feature selection is an important task. Certain features are [24] as follows:

1) Term presence, Term Frequency, Term Frequency–Inverse Document Frequency:

Term presence and Term frequency are the simplest form to use raw frequency of a term in a document [24] i.e. number of times term "t" occurs in document "d". We denote the frequency of a term in as "tf" and can give binary weights or logarithmically weight to show the importance of the feature. While inverse document frequency defines that how much information a term can provide for a classification of document i.e. whether a term is common or rare in all documents. While Term Frequency–Inverse Document Frequency is the product of two parts first is term presence and inverse document frequency. One can also apply weight to important features.

2) Part of Speech (POS):

POS is used to define the sematic meaning of word appearing in document i.e. adjective, noun, verb, etc.

3) Opinion word and Phrase:

Opinion words and phrases are used to represent opinion in document.

4) Negation:

Use of negation flips the opinion e.g. not happy is same as sad.

C. Feature selection Methods

There are two standard approaches for feature selection-first one is lexicon based approach and other is the statistical approach. Feature selection method treats a document as a collection of words (Bag of Words) or string of words, in particular, sequence in document. Few more available statistical feature selection methods are as follows:

1) Point wise Mutual Information:

The mutual information measure provides a formal way to model the mutual information between the features and the classes [24]. Information Theory defines this measure. Point Wise Mutual Information between word "w" and Class "C" is primarily determined by co-occurrences between them based on mutual independence is formulated as, F (w) ● Pc (W). Mutual information Mc (w) is defined as Eq. 1

$$Mc\,(w) = log\left(\frac{F(w).Pc(w)}{F(w).Pc}\right) \qquad (1)$$

The word w and class c have either positive or negative correlation depending upon the value of Mc (w). If Mc has a value greater than zero then correlation is positive and if less than zero then correlation is negative.

2) Chi-square(χ^2):

Let, n be the total number of documents in the collection, Pc (w) be the conditional probability of class c for documents that contain w, Pc be the global fraction of documents containing the class c, and F(w) be the global fraction of documents that contain the word w. Therefore, theχ^2 -statistic of the word between word w and class c is defined as Eq. 2 [24]

$$\chi^2 = \frac{n.F(w)^2.(Pc(w) - Pc)^2}{F(w).(1 - F(w)).Pc.(1 - Pc))} \qquad (2)$$

χ^2 and PMI are two different ways of measuring the correlation between terms and categories. χ^2 is better than PMI as it is a normalized value; therefore, these values are more comparable across terms in the same category [21].

D. Sarcasm Classification Techniques

Sentiment Classification techniques can be roughly divided into machine learning approach, lexicon based approach and hybrid approach [33]. Machine Learning approaches uses the different ML algorithms like SVM, Naive Bayes, Decision-tree, etc. based on lexical features and, is further divided into supervised machine learning methods that uses large labeled data set and unsupervised Machine learning techniques, which are used where it is hard to find labeled corpora to train the classifier. While Lexicon based approaches are further divided in two approaches dictionary based approach and corpus-based approach, they uses statistical and sentiment based methods for finding the sarcastic sentence.

E. Supervised Machine Learning Techniques

Supervised learning methods can be used when sufficient labeled training corpora is available. Sarcasm detection problem can be formed as: Given training document set D =

{d1, d2,d3,.. ,dn} with each document assigned one of the binary class label sarcastic or non-sarcastic. A Model M relates feature set of document to class labels. After that, given new document ‘d’, model M is used to predict class label for it. Some of the major methods are explained below.

1) Naive Bayes Classifier:

The Naive Bayes is one of the probabilistic classifiers, known as the generative classifier. Naive Bayes computes the posterior probability of class based on the distribution of word regardless of their position in document. It uses Bayes Theorem and Naive assumption that all features are independent to predict the probability of given feature belongs to the particular class. Equation is as follow,

$$P(label|features) = \frac{P(label)*P(f1|label)*...*P(fn|label)}{P(features)} \quad (3)$$

2) Maximum Entropy Classifier:

The Maximum Entropy Classifier (ME) is another probabilistic classifier that converts feature set into a vector using encoding using Vector a weight is calculated for each feature set, so it can be combined to determine class label. Alec Go et al. use ME classifier [1] for Sentiment Analysis.

3) Support Vector Machine :

Support Vector Machine (SVM) is the one of the linear classifiers used to define the linear separation of data points in search space that can best differentiate the different classes. Figure 3 represents an example of it with classes . In Figure 3 there are three hyper planes that can differentiate two classes but among them only Plane A can best separate them because A gives largest distance between any data points. One can achieve non-linear classification by using different SVM kernel techniques. SVM classifier is best suited for text data due to its sparse nature. Roberto González et al [16], Ellen Riloff et al. [19] and Ashwin Rajadesingan et al. [20] have used SVM for sarcasm detection in textual data.

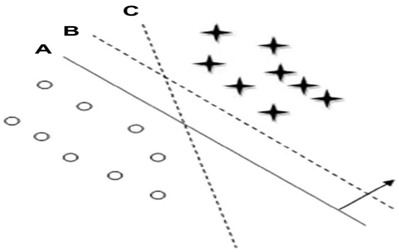


Fig. 3. Support Vector Machine Classifier

4) Conditional Random Field (CFR):

CRFs are a family of discriminative models first proposed by Lafferty et al. The commonest (linear-chain) CRFs follows Hidden Markov Models in that the next state depends on the current state (hence the ‘linear chain’ of dependency). CRFs generalize Logistic Regression for application to sequential data. CRFs defines the posterior probability of class label sequence ‘C’ given input observation sequence ‘F’, P (C|F). CRFs have been successfully applied for Named Entity Recognition and Part Of Speech tagging. Figure 4 explains CRFs.

Apart from these authors [18], [19], [20] have used Support Vector Machine with sequential minimal optimization, balanced winnow classifier, and Weighted nearest neighbour classification model for classification of sarcastic sentences.

F. Lexicon Based Approach

Lexicon based approach for sarcasm detection has been done in [19], [20]. In this method, opinion words are used to express the sentiment. Lexicon based approach is divided into two parts Dictionary Based Approach and Corpus Based Approach.

1) Dictionary Based Approach

In this approach, the set of opinion words is collected manually from best of knowledge. Then set size is increased by finding synonyms and antonyms from well know repository like Word Net. This approach will fail when finding context dependent opinion words. However, today for many languages Word Net like dictionaries are available. Ashwin Rajadesingan et al. [20] have taken help of behavioral model to represent sarcasm. In their work, they have used data set compiled by Warriner et al. [35] to calculate words affect score ,while word sentiment score is calculated using SentiStrenght [36].

2) Corpus Based Approach

To find context dependent opinion words. It relies on syntactic pattern - opinion words that occurs together in context to particular environment. While determining the semantic orientation of words WordNet-like dictionary is used . Work done by Ellen Riloff et al is based on this approach. They started by seed word ‘love’ to make their classifier learn, the positive words and negative situation for classification of sarcastic sentences [21]. Davidov et al. (2010) have created large number of Amazon review and Tweets to train their classifier to detect automatically sarcastic sentence [20].

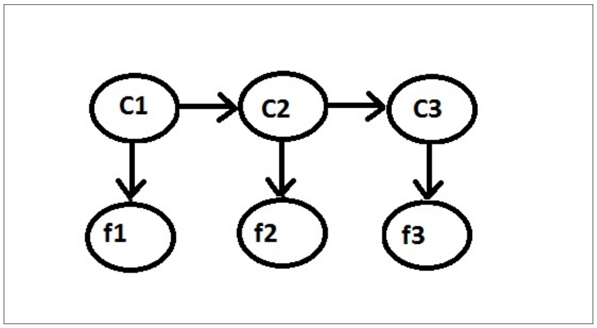
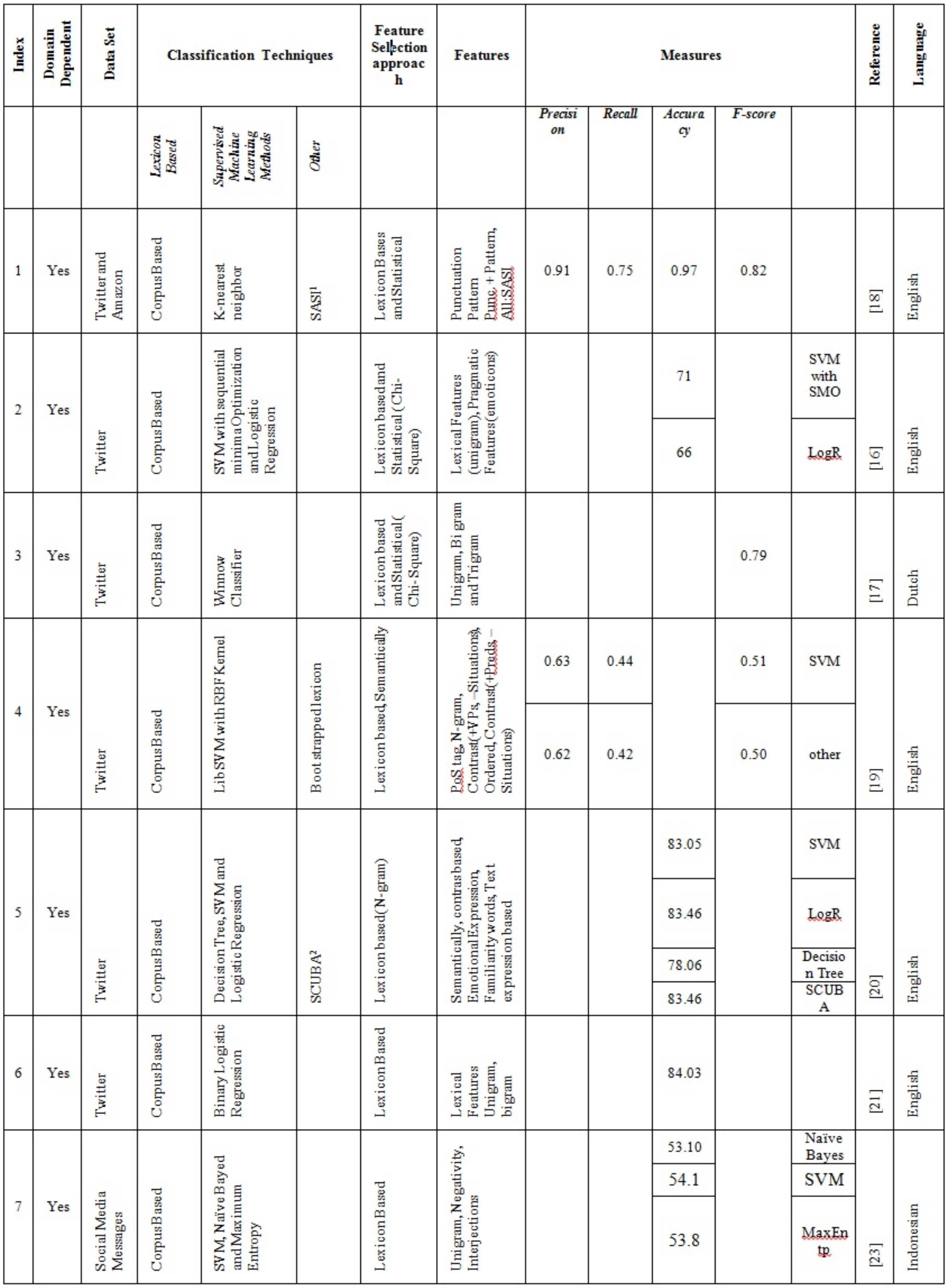


Fig. 4. Conditional Random Field

The concise summary, for major methods used for sarcasm detection is shown in Table1.

TABLE I. ANALYSIS OF DIFFERENT SARCASM DETECTION TECHNIQUES



Index	Domain Dependent	Data Set	Classification Techniques			Feature Selection approach	Features	Measures					Reference	Language
			Lexicon Based	*Supervised Machine Learning Methods*	*Other*			*Precision*	*Recall*	*Accuracy*	*F-score*			
1	Yes	Twitter and Amazon	CorpusBased	K-nearest neighbor	SASI[1]	LexiconBases and Statistical	Punctuation Pattern Punc + Pattern, All:SASI	0.91	0.75	0.97	0.82		[18]	English
2	Yes	Twitter	CorpusBased	SVM with sequential minima Optimization and Logistic Regression		Lexicon based and Statistical (Chi-Square)	Lexical Features (unigram), Pragmatic Features(emoticons)			71		SVM with SMO	[16]	English
										66		LogR		
3	Yes	Twitter	CorpusBased	Winnow Classifier		Lexicon based and Statistical(Chi-Square)	Unigram, Bi gram and Trigram				0.79		[17]	Dutch
4	Yes	Twitter	CorpusBased	LibSVM with RBF Kernel	Boot strapped lexicon	Lexicon based, Semantically	PoS tag, N-gram, Contrast(+VPs, –Situations), Ordered, Contrast(+Preds, –Situations)	0.63	0.44		0.51	SVM	[19]	English
								0.62	0.42		0.50	other		
5	Yes	Twitter	CorpusBased	Decision Tree, SVM and Logistic Regression	SCUBA[2]	Lexicon based(N-gram)	Semantically, contras based, Emotional Expression, Familiarity words, Text expression based			83.05		SVM	[20]	English
										83.46		LogR		
										78.06		Decision Tree		
										83.46		SCUBA		
6	Yes	Twitter	CorpusBased	Binary Logistic Regression		LexiconBased	Lexical Features Unigram, bigram			84.03			[21]	English
7	Yes	Social Media Messages	CorpusBased	SVM, Naïve Bayed and Maximum Entropy		LexiconBased	Unigram, Negativity, Interjections			53.10		Naïve Bayes	[23]	Indonesian
										54.1		SVM		
										53.8		MaxEntp		

a.SASI – Semi-Supervised Sarcasm Identification Algorithm

b.SCUBA– Sarcasm Classification Using Behavioural Modelling Approach

V. Experiment and results

After in-depth study of Sentiment Analysis in various languages including Hindi. We come across the fact that for Hindi language Unigram gives better performance with TFIDF than BiGram or NGram for SA. To reach our conclusion we have collected 150 positive sentences and 150 negative sentences from [2] and modified it to generate sarcastic sentence corpora, Then we trained the SVM classifier with 10X validation with simple Bag-Of-Words as features and TF-IDF as frequency measure of the feature. We have manually generated 25 sarcastic sentences test our model. Our experimental results show that accuracy of our system is 50%. We analyzed the results and found that simple Bag-Of-Words method is not sufficient for sarcasm detection but need more pragmatic and lexical feature. Few examples are discussed below.

e.g. मुझे नजरअंदाज होना पसंद है। (mujhee nazaarandaz hoonaa pasand haii) i.e. (I like being ignored). In above example two words “नजरअंदाज” (nazarandaz) and “पसंद” (pasand) are of opposite polarity namely negative and positive respectively ,and model predicts it as non-sarcastic (as total polarity becomes nill) and so gives it an overall sentiment of “neutral” but actually it is sarcastic sentence and its correct polarity should have been “negative”.

Similarly consider following example-

इसे निकाल देना चाहिऎ. ये बहोत आच्चा खिलाडी है। (ise nikal dena cha hie ,yeh bahut achha khiladi hai)

Here also if only bag-of-words feature is used, the model predicts it is “positive” sentiment, whereas by common sense it can be seen that this statement is sarcastic and hence should be considered “negative”. Thus we can conclude that we need some extra features to detect the sarcastic sentences like #tags, emoticons and such other informative features.

VI. CONCLUSION

Automatic Classification of Sarcastic sentence is one of major challenge in Sentiment Analysis. We have gone through prominent research article in this area for exploring various techniques for it. Our study gives overview of work carried out in area of Sarcasm Detection and will help naïve researchers of this field. Work carried out for sarcasm detection in non-Indian languages, has shown that help of extra features like #tags have improve the performance. In our study we have found that on an average 70% of articles for English language have used Unigram as feature and have achieved average accuracy around 68%. We have discussed various classification techniques their performance on textual data. We found that researchers have used hybrid technique i.e. usage of both Lexicon based and Machine Learning techniques together for classification. We also discussed various feature selection methods and features alternative available for sarcasm detection. We found that Twitter, Amazon and social media sites are primary and easily accessible data source.

Apart from English, sarcasm detection for Dutch and Indonesian language has also been carried out, which shows researcher’s interest in other languages. Sarcasm detection also depends on context of text available and needs more work to be carried out for better enhancements.

References

[1] Go, Alec, Richa Bhayani, and Lei Huang. "Twitter sentiment classification using distant supervision." CS224N Project Report, Stanford 1 (2009): 12.

[2] Joshi, Aditya, A. R. Balamurali, and Pushpak Bhattacharyya. "A fall-back strategy for sentiment analysis in Hindi: a case study." Proceedings of the 8th ICON (2010).

[3] Jiang, Long, et al. "Target-dependent twitter sentiment classification." Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics, 2011.

[4] Pak, Alexander, and Patrick Paroubek. "Twitter as a Corpus for Sentiment Analysis and Opinion Mining." LREC. Vol. 10. 2010.

[5] Hu, Minqing, and Bing Liu. "Mining and summarizing customer reviews." Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2004.

[6] Wilson, Theresa, Janyce Wiebe, and Paul Hoffmann. "Recognizing contextual polarity in phrase-level sentiment analysis." Proceedings of the conference on human language technology and empirical methods in natural language processing. Association for Computational Linguistics, 2005.

[7] Read, Jonathon. "Using emoticons to reduce dependency in machine learning techniques for sentiment classification." Proceedings of the ACL student research workshop. Association for Computational Linguistics, 2005.

[8] Pang, Bo, and Lillian Lee. "Opinion mining and sentiment analysis." Foundations and trends in information retrieval 2.1-2 (2008): 1-135.

[9] Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. "Thumbs up?: sentiment classification using machine learning techniques." Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. Association for Computational Linguistics, 2002.

[10] Narr, Sascha, Michael Hulfenhaus, and Sahin Albayrak. "Language-independent twitter sentiment analysis." Knowledge Discovery and Machine Learning (KDML), LWA (2012): 12-14.

[11] Maynard, Diana, Kalina Bontcheva, and Dominic Rout. "Challenges in developing opinion mining tools for social media." Proceedings of the@ NLP can u tag# usergeneratedcontent (2012): 15-22.

[12] Dave, Kushal, Steve Lawrence, and David M. Pennock. "Mining the peanut gallery: Opinion extraction and semantic classification of product reviews." Proceedings of the 12th international conference on World Wide Web. ACM, 2003.

[13] Nigam, Kamal, John Lafferty, and Andrew McCallum. "Using maximum entropy for text classification." IJCAI-99 workshop on machine learning for information filtering. Vol. 1. 1999.

[14] Turney, Peter D. "Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews." Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, 2002.

[15] Cristianini, Nello, and John Shawe-Taylor. An introduction to support vector machines and other kernel-based learning methods. Cambridge university press, 2000.

[16] González-Ibánez, Roberto, Smaranda Muresan, and Nina Wacholder. "Identifying sarcasm in Twitter: a closer look." Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2. Association for Computational Linguistics, 2011.

[17] Liebrecht, C. C., F. A. Kunneman, and A. P. J. van den Bosch. "The perfect solution for detecting sarcasm in tweets# not." (2013).

[18] Davidov, Dmitry, Oren Tsur, and Ari Rappoport. "Semi-supervised recognition of sarcastic sentences in twitter and amazon." Proceedings of

the Fourteenth Conference on Computational Natural Language Learning. Association for Computational Linguistics, 2010.

[19] Riloff, Ellen, et al. "Sarcasm as Contrast between a Positive Sentiment and Negative Situation." EMNLP. 2013.

[20] Rajadesingan, Ashwin, Reza Zafarani, and Huan Liu. "Sarcasm detection on Twitter: A behavioral modeling approach." Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. ACM, 2015.

[21] Bamman, David, and Noah A. Smith. "Contextualized Sarcasm Detection on Twitter." Ninth International AAAI Conference on Web and Social Media. 2015

[22] Smith, Phillip, et al. "Sentiment Analysis: Beyond Polarity." (2011).

[23] Lunando, Edwin, and Ayu Purwarianti. "Indonesian social media sentiment analysis with sarcasm detection." Advanced Computer Science and Information Systems (ICACSIS), 2013 International Conference on. IEEE, 2013.

[24] Medhat, Walaa, Ahmed Hassan, and Hoda Korashy. "Sentiment analysis algorithms and applications: A survey." Ain Shams Engineering Journal 5.4 (2014): 1093-1113.

[25] Tsytsarau, Mikalai, and Themis Palpanas. "Survey on mining subjective data on the web." Data Mining and Knowledge Discovery 24.3 (2012): 478-514.

[26] Nadkarni, Prakash M., Lucila Ohno-Machado, and Wendy W. Chapman. "Natural language processing: an introduction." Journal of the American Medical Informatics Association 18.5 (2011): 544-551.

[27] Aggarwal, Charu C., and ChengXiang Zhai. Mining text data. Springer Science & Business Media, 2012.