Third International Conference on "Materials, Computing and Communication Technologies"
in association with International Journal of Scientific Research in Science,
Engineering and Technology
Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com)

Sarcastic and Non Sarcastic Word Detection in Social Media

Alagu Sundari N, Prof. S. Suresh Thangakrishnan
Department of Computer Science and Engineering, Einstein College of Engineering, Seethaparpanallur,
Tirunelveli, Tamil Nadu, India

ABSTRACT

Sarcasm is a type of sentimental analysis where people express their feeling either in sarcastic or non- sarcastic text through social media. Classification of sarcastic sentence into positive and negative sentiments has been identified as a difficult problem. In this paper, the main objective of this is to find out the data in sarcastic or non-sarcastic word. The amount of data retrieved from the public social media like YouTube, Facebook, and Twitter etc..,. Sarcasm is a special kind of sentiment that comprises of words that what you really want to say (e.g., Insult someone, funny, and to shoe irritation).People often express it through the use of heavy tonal stress or certain gestural clues like rolling of eyes.

Key word: Sarcastic, Non-Sarcastic, Social Media, Sentiments, Social Media

I. INTRODUCTION

The process of evaluating data using analytical and logical reasoning to examine each component of the data provided. Data from various sources is gathered, reviewed, and then analysed to form some sort of finding or conclusion. There are various specific data analysis method, some of which include data mining, text analysis and data visualization. R is perhaps one of the most powerful and most popular platforms for statistical programming and applied machine learning. Sarcasm detection is the task of correctly labelling the text as "Sarcastic" or "Non Sarcastic" word. It is challenging task due to the lack of intonation and facial expression in text. The use of sarcasm is prevalent across the all social media. Sarcasm analysis, being one of the toughest challenges in Natural Language Processing (NLP) has become a hot topic of research these days. NLP is one of the important domains in artificial intelligence. It acts as a platform between the computer and human languages. It helps in making the machine understand, analyse and interpret the data. It helps in querying the datasets and provide an answer. It helps in not only understand the text or speech but also the context behind it. It work for structure and unstructured data. Sarcasm is one of the leading challenges faced in Sentimental Analysis. Sarcasm is an indirect manner of conveying a message. It is basically a bitter expression which is conveyed. Sarcasm can be expressed in many ways. It can be expressed in speech and text. Sarcasm can be conveyed through various ways like a direct conversion, speech, text, etc. In direct conversation, facial expression and body gestures provide the hint of sarcasm. In the speech, sarcasm can be inferred if there is any changes in tone.

Machine Learning:

Machine learning is a subset of artificial intelligence that provides computers with the ability to learn without being explicitly programmed. Machine learning focuses on the development of computer programs, those can teach themselves to grow and change when exposed to new data. It considers the prediction problem as a problem of supervised learning problem, where I have to infer from historical data the possibly nonlinear dependence between the input (past embedding vector) and the output (future value).The area of Machine Learning deals with the design of programs that can learn rules from data, adapt to changes, and improve performance with experience. In addition to being one of the initial dreams of Computer Science, Machine Learning has become crucial as computers are expected to solve increasingly complex problems and become more integrated into our daily lives. Writing a computer program is a bit like writing down instructions for an extremely literal child who just happens to be millions of times faster than you. Yet many of the problems we now want computers to solve are no longer tasks we know how to explicitly tell a computer how to do. These include identifying faces in images, autonomous driving in the desert, finding relevant documents in a database (or throwing out irrelevant ones, such as spam email), finding patterns in large volumes of scientific data, and adjusting internal parameters of systems to optimize performance. That is, we may ourselves be good at identifying people in photographs, but we do not know how to directly tell a computer how to do it. Instead, methods that take labelled training data (images labelled by who is in them, or email messages labelled by whether or not they are spam) and then learn appropriate rules from the data, seem to be the best approaches to solving these problems. NLP is a branch of data science that consists of systematic processes for analysing, understanding, and deriving information from the text data in a smart and efficient manner. Here used package of R studio to apply NPL in order to do the following tasks. Removal of new lines and tabs, removal of punctuations, separating hash tagged words, tokenizing the inputs, removal of stop wards

II. RELATED WORK

R is a system for statistical analyses and graphics created by Ross Ihaka and Robert Gentleman1. R is both a software and a language considered as a dialect of the S language created by the AT&T Bell Laboratories. S is available as the software S-PLUS commercialized by Insightful2. There are important differences in the designs of R and of S: those who want to know more on this point can read the paper by Ihaka & Gentleman (1996) or the R-FAQ3, a copy of which is also distributed with R. R is freely distributed under the terms of the GNU General Public License 4; its development and distribution are carried out by several statisticians known as the R Development Core Team. R is available in several forms: the sources (written mainly in C and some routines in FORTRAN), essentially for UNIX and Linux machines, or some pre-compiled binaries for Windows, Linux, and Macintosh. The files needed to install R, either from the sources or from the pre-compiled binaries, are distributed from the internet site of the Comprehensive R Archive Network (CRAN) 5 where the instructions for the installation are also available. Regarding the distributions of Linux (Debian . . .), the binaries are generally available for the most recent versions; look at the CRAN site if necessary. R has many functions for statistical analyses and graphics; the latter are visualized immediately in their own window and can be saved in various formats (jpg, png, bmp, PS, pdf, emf, pictex, xfig; the available formats may depend on the operating system). The results from a statistical analysis are displayed on the screen, some intermediate results (P-values, regression coefficients, residuals . . .) can be saved, written in a file, or used in subsequent analyses. The R language allows the user, for instance, to program loops to successively analyse several data sets. It is also possible to combine in a single program different statistical functions to perform more complex analyses. The R users may benefit from a large number of programs written for S and available on the internet6, most of these programs can be used directly with R. At first, R could seem too complex for a non-specialist. This may not be true actually. In fact, a prominent feature of R is its flexibility. Whereas a classical software displays immediately the results of an analysis, R stores these results in an “object”, so that an analysis can be done with no result displayed. The user may be surprised by this, but such a feature is very useful. Indeed, the user can extract only the part of the results which is of interest. For example, if one runs a series of 20 regressions and wants to compare the different regression coefficients, R can display only the estimated coefficients: thus the results may take a single line, whereas a classical software could well open 20 results windows. We will see other examples illustrating the flexibility of a system such as R compared to traditional software’s. The term “environment” is intended to characterize it as a fully planned and coherent system, rather than an incremental accretion of very specific and inflexible tools, as is frequently the case with other data analysis software. R is very much a vehicle for newly developing methods of interactive data analysis. It has developed rapidly, and has been extended by a large collection of packages. However, most programs written in R are essentially ephemeral, written for a single piece of data analysis. The fact that R is a language may deter some users who think “I can’t program”. This should not be the case for two reasons. First, R is an interpreted language, not a compiled one, meaning that all commands typed on the keyboard are directly executed without requiring to build a complete program like in most computer languages (C, FORTRAN, Pascal . . .). Second, R’s syntax is very simple and intuitive. For instance, a linear regression can be done with the command lm(y ~ x) which means “fitting a linear model with y as response and x as predictor”. In R, in order to be executed, a function always needs to be written with parentheses, even if there is nothing within them (e.g., ls ()). If one just types the name of a function without parentheses, R will display the content of the function. In this document, the names of the functions are generally written with parentheses in order to distinguish them from other objects, unless the text indicates clearly so. When R is running, variables, data, functions, results, etc., are stored in the active memory of the computer in the form of objects which have a name. The user can do actions on these objects with operators (arithmetic, logical, comparison . . .) and functions (which are themselves objects).

III. METHODOLOGY & DESIGN

Existing System:

For a set of data, the intention is to classify each data depending on whether it is sarcastic or not. Hence for each comment, the extraction of a set of feature, referred as a training set and uses machine learning algorithms to perform the classification was used. To collect sarcastic tweets, it queries the incorporate the hashtag

„„„#sarcasm‟‟‟. In total, collection of 3000 comments with the hashtag „„„#sarcasm‟‟‟, which was cleaned up by removing the noisy and irrelevant ones alone. As for non-sarcastic tweets, they collected tweets dealing with different topics and made sure they have some emotional content. The whole study revolved around 3 data sets. First set contains 3000 data, half of them are sarcastic, and the other half are not. The tweets on this data set are manually checked and segregated depending on their level of sarcasm from 0 (highly non-sarcastic) to 1(highly sarcastic).For the second one no manual check is done, which makes it a very noisy data set. The third one, all comments are manually monitored and classified as sarcastic and non-sarcastic. This set will serve as a test set, and will be used to evaluate.

Proposed System:

A randomly collected set of data is taken, each one of them depending on whether it is sarcastic or not. Therefore, from each data, extract a set of features, refer to a training set and use machine learning algorithms to perform the classification. The features are extracted in a way that makes use of different components of the data, and used the sarcasm. The set of data on which run experiments is checked and annotated manually. The proposed system can detect sarcasm based on the text.

System Design:

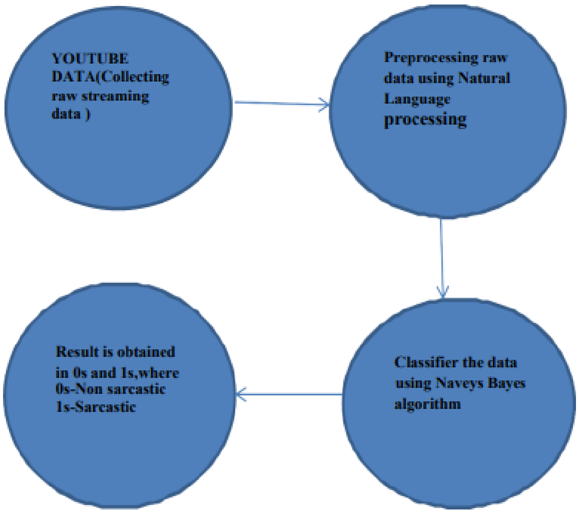
YOUTUBE DATA(Collecting raw streaming data)
Preprocessing raw data using Natural Language processing
Classifier the data using Naveys Bayes algorithm
Result is obtained in 0s and 1s,where 0s-Non sarcastic 1s-Sarcastic

Figure 2: Proposed system design

Dataset:

The data will gather to do the project with, will likely to be live streaming data that I gather for a period of a few days or weeks until I have a sufficiently large amount to train our system with. However YouTube data is problematic, while the data is readily available there is little to no context due to the short messages. In order to gain context with data we need to look at replies, past tweets, and the user’s profile. While these may be possible it is a more advanced option that I hope to be able to get to by the end of the project, but using and gathering the context for each data is more of a stretch goal. Ideally I will be able to attain a data set that has more context in the surrounding text and does not require specific background of the actors. To narrow the scope, using data initially, I will select #sarcasm, #sarcastic, etc. as well as other hashtags that allow us to tailor our system to a specific niche area to focus upon. This focus will make our system less generalizable but, in theory, be more accurate with that particular data set. If I am able to attain a sufficiently high accuracy with a niche focus then testing the system on a more general data selection would be the next goal. My data will be tagged as sarcasm or not sarcasm so I will be using primarily supervised learning techniques.

Corpus:

Corpus is a large collection of texts. It is a body of written or spoken material upon which a linguistic analysis is based. The plural form of corpus is corpora. Some popular corpora are British National Corpus (BNC), COBUILD/Birmingham Corpus, and IBM/Lancaster Spoken English Corpus. Monolingual corpora represent only one language while bilingual corpora represent two languages. European Corpus Initiative (ECI) corpus is multilingual having 98 million words in Turkish, Japanese, Russian, Chinese, and other languages. The corpus may be composed of written language, spoken language or both. Spoken corpus is usually in the form of audio recordings. A corpus may be opened or closed. An opened corpus is one which does not claim to contain all data from a specific area while a closed corpus does claim to contain all or nearly all data from a particular field. Historical corpora, for example, are closed as there can be no further input to an area. The main structure for managing documents in tm is a so- called Corpus, representing a collection of text documents. ... The default implementation is the so- called VCorpus (short for Volatile Corpus) which realizes a semantics as known from most R objects: corpora are R objects held fully in memory

Bag of Words:

The Bag of words is a way of representing text data when modelling text with machine learning algorithm. The Bag of words model is simple to understand and implement and is seen great success in problem such as language modelling and documentation classification.

Document Term Matrix:

A document-term matrix or term-document matrix is a mathematical matrix that describes the frequency of terms that occur in a collection of documents. In a document-term matrix, rows correspond to documents in the collection and columns correspond to terms. There are various schemes for determining the value that each entry in the matrix should take.. They are useful in the field of natural language processing.

IV. RESULT AND DISCUSSION

My primary goal is to analyse the data whether it is in sarcastic or non-sarcastic. Computational detection of sarcasm is seen attention from the sentiment analysis community in the past few years. Sarcasm is an interesting problem for sentiment analysis because surface sentiment of words in sarcastic text may be different from the implied sentiment. For Example, “Being stranded in traffic is the best way to start a week” is a sarcastic sentence because the surface sentiment of the word “best” (positive) is different from the implied sentiment of the sentence (negative), considering remaining portion of the text. Proposed an algorithm to understand a sentence or data is sarcastic or not. From the dataset I have acquired the data, I try to train our model so that test live streaming data from social media to detect whether they are sarcastic or not. This will result in obtaining accurate sentiment analysis and opinion mining. It could contribute to enhanced automated feedback systems in the context of customer based sites.

V. CONCLUSION

Social media is an informal means of communication so with the absence of complete background knowledge it is very hard to predict the exact nature of tweets. Further short forms, abbreviated words, short-form words, emoji's can add a degree of vagueness, inconsistency in the results. Sarcasm detection and analysis in social media provides invaluable insight into the current public opinion on trends and events in real time. Various classification and NLP are implemented to find out the best model. Navie Baye's Classification algorithm are used. Sarcasm can be found in variety of topics, it need to sample a large data. Finding new features relevant to sarcasm is a critical step for improving sarcasm detection and expect to see that some sort of contextual clues (feature types such as sentiment contrast, order of words, etc.) to play a big role in the sarcastic nature of a sentence.

VI.REFERENCES

[1]. D. Bamman, N. Smith Copyright © 2015, Association for the Advancement of Artificial Intelligence.

[2]. A Closer Look R. González-Ibáñez S. Muresan N. Wacholder HLT '11 Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Volume 2 581-586 Association for Computational Linguistics Stroudsburg, PA, USA ©2011

[3]. Lexical influences on the perception of sarcasm, R. J. Kreuz and G. M. Caucci, in Proceedings of the Workshop on computational approaches to Figurative Language, ACL, pp. 1--4, 2007.

[4]. Parsing-based Sarcasm Sentiment Recognition in Twitter Data S.Kumar Bharti K. Sathya S. Kumar Jena ASONAM '15 Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015 Pages 1373-1380 ACM New York, NY, USA ©2015

[5]. Recognition of Sarcasm in Tweets Based on Concept Level Sentiment Analysis and Supervised Learning Approaches P. Tungthamthiti K. Shirai M. Mohd

[6]. Sarcasm as Contrast between a Positive Sentiment and Negative Situation E.Riloff, A. Qadir, P. Surve, L.De Silva, N.Gilbert, R. Huang EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference Association for Computational