JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Class imbalanced detection of sarcasm in social networking websites based on minority up-weighting

[1]Rahul Sen, [2]Mehtab Alam Khan, [3]Md. Tarik Aziz
[1,2,3]Undergraduate Student,
[1,2,3]Computer Science and Engineering,
[1,2,3]University of Engineering & Management, Kolkata, India

Abstract: Over the past few decades, the evolution of machine learning has been tremendous, which gave solutions to many real-world problems and still making lives more comfortable as well as innovative. One of the applications of machine learning would be to detect sarcasm in social media like Twitter, Facebook etc., which will, in turn, give us insight into a topic trending in social networking websites. Thus this mining of people's opinions would be of use to make decisions in many ways. But due to the highly imbalanced classes of social media data, it is often difficult to get the accuracy that is desired. So, to deal with this type of dataset, in this study, synthetic minority oversampling based methods are proposed. The main focus of this article is minority up-weighting, hence, two methods namely KMeansSMOTE and BorderlineSMOTE algorithm are used along with classifiers like Bernoulli Naive Bayes (BNB), Multilayer Perceptron (MLP) and Decision Tree Classifier (DTC) to get better accuracy while dealing with imbalanced dataset. Thus this article contains an analysis of imbalanced classification in the detection of sarcasm in social media through minority up-weighting techniques.

Keywords: Machine learning, sarcasm detection, Imbalanced class, minority up-weighting.

1. INTRODUCTION

Now a days, people are always connected through social networking platforms like Twitter, which is a microblogging website and has been a great means to exchange information and opinions through the internet. That is why for this study and analysis, the data from Twitter is considered. People continuously give their opinion on trending topics and share it with the rest of the world through this platform, like about a new movie [3], natural disaster [1] etc. Along with reviews and critics, people often share their sarcastic views on various matter irrespective of the underlying seriousness. Sarcasm, thus can be stated as a positive statement having a negative meaning [2] which can also be regarded as mockery towards a particular idea or thing.

Detection of sarcasm is based on the algorithms of supervised learning [7]. Mining these text data from Twitter to detect sarcasm is a widely accepted approach for getting insight on a topic and also for considering people's opinions. A person can comment sarcastically on any topic including serious matters. In this article [2], it is discussed that one of the reasons for a person to give a sarcastic comment is because of jealously towards a matter, which is based on human nature. And sometimes people comment sarcastically because of the stupidity of that particular matter. But the class imbalance in the dataset from the social networking site makes it difficult to achieve the desired accuracy. In this article, we are going to address this imbalanced classification for the detection of sarcasm by using Synthetic Minority Oversampling techniques along with different classifiers for proving credibility.

2. LITERATURE SURVEY

Over the last decade, research on sarcastic text mining and detection has advanced a lot especially in the field of natural language processing (NLP). In article [9], the writer detected sarcasm by displaying a fascinating experiment based on the flow of sentiment from positive to negative and vice versa. In article [6], the experiment illustrated that better accuracy can be achieved by incorporating extra information from the tweet text like the aur, audience details etc. There has also been a study of automatic detection of sarcasm with a detailed analysis of different classifiers. Interestingly, this experiment was also used to detect the nastiness in the text [4]. A study has also been conducted where a model based on Hadoop was used for the detection of sarcasm [5]. In the article [8], an extensive exhibit of comparison of the different classifiers were recorded for sarcasm detection. Sarcasm detection based on sentence completion has also been illustrated in the article [10], which increased the accuracy to a large extend. In the article [15], the detection of sarcasm was caused by utilising a semi-supervised sarcasm identification algorithm which is also known as SASI. There has also been a study depicting the relation between sentiment analysis and sarcasm detection, and how these two are related [11]. In article [13], the study proved that for huge training data set, detection of sarcasm can be achieved by recognising patterns in the parts of speech in an efficient way and in that analysis it has achieved an astonishing accuracy over other methods. In article [12], SHRINK algorithm was introduced for taking care of the overlapping region of the minority class. Not even in English, the analysis of sarcasm detection in other languages like Hindi and Czech has also been studied thoroughly. In article [16], sarcasm was detected from text from tweets that were written in Hindi. In [17], the accuracy of detection of sarcasm in tweets written in Hindi was seen to increase. Following the trend Ptáček, in [14] suggested an extraordinary approach using SVM classifier to detect sarcasm in Czech language text.

Based on these studies and analysis of different methods of detection of sarcasm, this paper proposed a thorough study of the effect of imbalanced class for detection of sarcasm in social networking websites especially in Twitter by introducing minority up-weighting algorithms and comparing those with different well-known classifier.

3. PROPOSED METHODOLOGY

The proposed approach in this paper is to build a machine learning model which will mitigate the effect of imbalance classification. The dataset used in the paper has fewer non-sarcastic tweets than sarcastic tweets, which raised the imbalance problem for this experiment. So, for developing this model, the first step is to extract the data from the source. In this experiment, the data from Twitter is considered and thus a Corpus is created. After the creation of the Corpus is done, vectorization is performed and features are extracted for the analysis. After extracting the feature, the minority class is oversampled and finally, the model is trained with different classification algorithms. The performance is then recorded with tabular form. In figure 1, a detailed flow chart is depicting the proposed method.

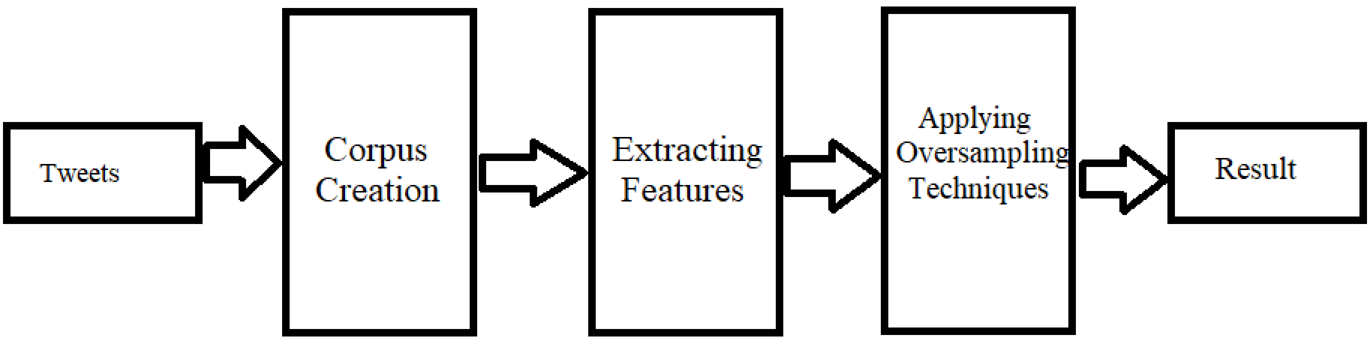

Fig. 1 Flow Diagram of the Proposed Methodology

3.1. CORPUS CREATION

For the analysis of imbalance classification, the dataset which is constructed is extracted from Twitter. Out of all the social media platforms, Twitter data is considered due to its popularity in opinion sharing, thereby giving the perfect set of data for opinion mining. The dataset contains tweets on various topics. This dataset contains both sarcastic as well as non-sarcastic tweets. However, the number of sarcastic tweets are far more than non-sarcastic tweets. Out of the total 52,000 tweets considered 38,000 tweets are sarcastic whereas 13,000 tweets are non-sarcastic. So, clearly, the ratio in the dataset shows that the non-sarcastic data is the minority class. After the extraction of the data, it is also cleaned by removing emoji, hashtags, punctuation and other unwanted characters. Thus we create the corpus for the model used in this experiment. In figure 2, the bar-graph depicts the imbalance in the classes.

3.2. FEATURE EXTRACTION

Now after the corpus is created, the next step is to extract the features. So, for the experiment, the unigram feature is considered. Also, the classifier of the model cannot handle text, thus the text data is converted into a numerical form with the help of the vectorization process. In this study, the Term Frequency-Inverse Document Frequency method is used for vectorization. In simple language, this method depicts how important a word is in a corpus. This method has two-part, the Term Frequency depicts the number of occurrences of a particular word, whereas the Inverse Document Frequency depicts the number of occurrences of a particular word in the whole corpus which is considered. Thus this method of vectorization returns a value in a sparse matrix for each word. This is how, this method converts the text into a numerical format.

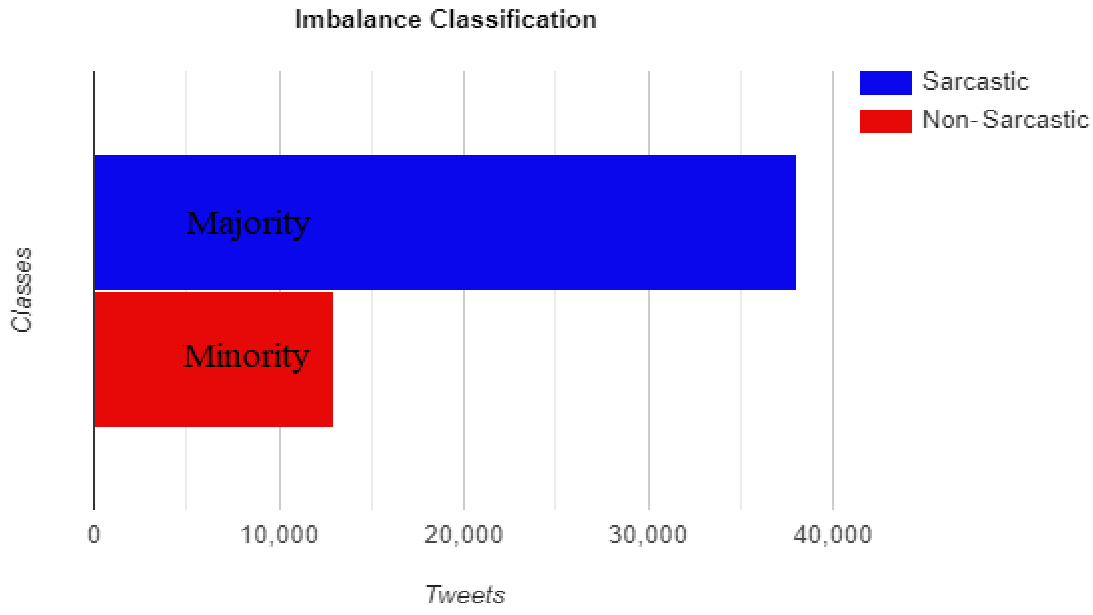

Fig. 2 Insight of the dataset used in the experiment

3.3. OVERSAMPLING THE MINORITY CLASS

After the vectorization process is completed, the next step will be to overcome the class imbalance problem in the model with the oversampling techniques. In simple language, oversampling means introducing manipulated or artificial data to the corpus so that the minority and the majority class is at the same level. In this section, two methods are proposed namely KMeansSMOTE and BorderlineSMOTE algorithm for oversampling. The previous section has depicted that there is a clear imbalance between the two classes. Now to balance the two classes equally, either up-weighting or down-weighting methods can be used. In the corpus used, the sarcastic data is the majority class and non-sarcastic data is the minority part. In the process of the down-weighting method the majority class is under-sampled to equalise with the minority class whereas, in the up-weighting method, the minority class is oversampled to equalise with the majority class. In the main focus of this paper is up-weighting the minority class, thus the above-mentioned algorithm is used along with classifiers like Bernoulli Naive Bayes (BNB), Multilayer Perceptron (MLP) and Decision Tree Classifier (DTC) to overcome or mitigate the effect of class imbalance.

4. EXPERIMENTAL RESULTS

This section of the paper deals with the experimental result of the model that is constructed. The model is imposed with the algorithm which was discussed earlier, that is KMeansSMOTE and BorderlineSMOTE. Each algorithm is then passed with a different classifier like Bernoulli Naive Bayes (BNB), Multilayer Perceptron (MLP) and Decision Tree Classifier (DTC) and also increased the oversampling rate by 10% in each step. The accuracy is then measured with the help of the F1 Score (F1) and Precision (Pre) scores. The accuracy is recorded in tabular form for comparison of the two algorithm considered. In the KMeansSMOTE algorithm with Multilayer Perceptron (MPL) classifier, when the oversampling quantity is 100%, then F1 value was 0.92 whereas the pre value was recorded to be 0.91. Similarly, with BorderlineSMOTE the F1 score was 0.92 and Pre value was 0.92 as well. Now under the Decision Tree Classifier (DTC) classifier, for 100% oversampling quantity both F1 score was 0.91 and Pre value was 0.92 for KMeansSMOTE whereas for BorderlineSMOTE both the F1 and Pre score was recorded to be 0.92. Finally under Bernoulli Naive Bayes (BNB) Classifier when the oversampling rate was 100%, for the KMeansSMOTE algorithm the F1 score was 0.93 and Pre was 0.91 whereas for BorderlineSMOTE the value for F1 and pre was recorded to be 0.91 and 0.92 respectively. Figure 3, 4, 5 shows an in-depth result of the experiment in tabular form.

KMeansSMOTE		
OVERSAMPLING QUANTITY	F1	PRE
10	0.90	0.91
20	0.91	0.91
30	0.92	0.90
40	0.91	0.91
50	0.92	0.92
60	0.91	0.92
70	0.91	0.91
80	0.91	0.92
90	0.92	0.92
100	0.92	0.91

BorderlineSMOTE		
OVERSAMPLING QUANTITY	F1	PRE
10	0.90	0.91
20	0.89	0.90
30	0.91	0.92
40	0.89	0.90
50	0.91	0.91
60	0.92	0.92
70	0.91	0.91
80	0.91	0.92
90	0.92	0.92
100	0.92	0.92

Fig. 3 Performance of the two algorithm under MLP Classifier

KMeansSMOTE		
OVERSAMPLING QUANTITY	F1	PRE
10	0.91	0.90
20	0.92	0.92
30	0.94	0.94
40	0.93	0.94
50	0.91	0.91
60	0.92	0.93
70	0.92	0.92
80	0.91	0.92
90	0.90	0.91
100	0.91	0.92

BorderlineSMOTE		
OVERSAMPLING QUANTITY	F1	PRE
10	0.90	0.91
20	0.91	0.91
30	0.89	0.90
40	0.90	0.89
50	0.90	0.92
60	0.89	0.90
70	0.91	0.90
80	0.91	0.91
90	0.90	0.91
100	0.91	0.92

Fig. 4 Performance of the two algorithm under DTC Classifier

KMeansSMOTE		
OVERSAMPLING QUANTITY	F1	PRE
10	0.91	0.93
20	0.91	0.92
30	0.92	0.93
40	0.94	0.93
50	0.92	0.93
60	0.91	0.93
70	0.91	0.92
80	0.93	0.92
90	0.91	0.92
100	0.93	0.91

BorderlineSMOTE		
OVERSAMPLING QUANTITY	F1	PRE
10	0.93	0.93
20	0.94	0.93
30	0.92	0.93
40	0.91	0.92
50	0.93	0.90
60	0.90	0.91
70	0.93	0.92
80	0.92	0.92
90	0.92	0.92
100	0.91	0.92

Fig. 5 Performance of the two algorithm under BNB Classifier

5. CONCLUSION

So, in this article, a thorough research was conducted for finding the effects of imbalance classification in the detection of sarcasm from social networking sites. For this experiment, a dataset was considered where the corpus was created from the data extracted from Twitter to making the machine learning model. The non-sarcastic data was the minority class whereas the sarcastic data was the majority class. Then the features are extracted from the corpus. After extracting the features, oversampling techniques are implemented on the model along with the classifier to find the F1 and Pre score. Thus a comparison was made for the two oversampling techniques considered for this experiment under different classifiers in tabular form.

REFERENCES

[1] Tan S, "An effective refinement strategy for KNN text classifier", 2006, Expert Systems Application-30: 290–298.

[2] Ragavan T, Priyanthan P, Gokulakrishnan B, Prasath N, Perera, "An Opinion mining and sentiment analysis on a Twitter data stream", 2012, International Conference on Advances in ICT for Emerging Regions, pp 182-188.

[3] Vikas Khullar V, Anjuman P, "Sentiment classification on big data using Naïve Bayes and logistic regression", 2017, International Conference on Computer Communication and Informatics, pp 1–5.

[4] Joshi A, Bhattacharyya P, Carman MJ, "Sarcasm detection using contextual incongruity, Investigations in computational sarcasm", 2018, Cognitive systems monographs, vol 37. Springer.

[5] Bharti SK, Vachha B, Pradhan RK, Babu KS, Jena SK, "Sarcastic sentiment detection in tweets streamed in real time: a big data approach", 2016, Digital Communications and Network 2(3) pp 108–121.

[6] Smith NA, Bamman D, "Contextualized sarcasm detection on twitter", 2014, Proceedings of the Ninth International AAAI conference on Web and Social Media, pp 574–577.

[7] Bhattacharyya P, Joshi A, Carman MJ, "Sarcasm detection using contextual incongruity", 2018, Investigations in computational sarcasm. Cognitive systems monographs, vol 37. Springer.

[8] Bharti SK, Pradhan R, Babu KS, Jena SK, "Sarcasm analysis on twitter data using machine learning approaches", 2017, Lecture Notes in Social Networks. Springer.

[9] Filatova E, "Irony and sarcasm: corpus generation and analysis using crowdsourcing", 2012, Proceedings of the Eighth International Conference on Language Resources and Evaluation, European Languages Resources Association, pp 392–398.

[10] Japkowicz N, "The class imbalance problem: significance and strategies", 2000, Proceedings of the 2000 International Conference on Artificial Intelligence: special track on inductive learning.

[11] Bouazizi M, Ohtsuki T, "Opinion mining in Twitter: How to make use of sarcasm to enhance sentiment analysis", 2015, IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp 1594–1597.

[12] Kubat M, Matwin S, "Addressing the curse of imbalanced training sets: one sided selection", 1997, Proceedings of the fourteenth international conference on machine learning, pp 179–186.

[13] Bouazizi M, Otsuki Ohtsuki T, "A pattern-based approach for sarcasm detection on twitter", 2016, IEEE Access 4: 5477–5488.

[14] Ptáček T, Habernal I, Hong J, "Sarcasm detection on Czech and english twitter", 2014, Proceedings of COLING, The 25th international conference on computational linguistics: technical papers, pp 213–223.

[15] Davidov D, Tsur O, Rappoport A, "Semi-supervised recognition of sarcastic sentences in Twitter and Amazon", 2010, Proceedings of the Fourteenth Conference on Computational Natural Language Learning Association for Computational Linguistics, pp 107–116.

[16] Bharti SK, Babu KS, Raman R, "Context-based sarcasm detection in Hindi tweets", 2017, Ninth International Conference on Advances in Pattern Recognition (ICAPR), pp 1–6.

[17] Bharti SK, Sathya BK, Jena SK, "Harnessing online news for sarcasm detection in hindi tweets", 2017, Pattern recognition and machine intelligence 2017, Lecture notes in computer science. Springer.