A Novel Balancing Technique with TF-IDF Matrix for Short Text Classification to Detect Sarcasm

Rajshree Singh[1], Dr. Reena Srivastava[2]

[1]Research Scholar, [2]Dean

School Of Computer Application, Babu Banarasi Das University, Lucknow

Abstract

Sarcasm is the use of statements that imply the opposite of what they represent. The study of numerous forms of language expressions, including humor, irony, sarcasm, hatred, enmity, etc., has developed into a profitable research area with the quantity of data available from social media. The automatic detection of particular expressions is a highly debated topic in Natural Language Processing (NLP). This study aims to locate one of the most commonly used linguistic expressions throughout social media platforms (SMPs), "sarcasm." This larger, class-balanced dataset is used to meet the model's requirements (Swami et al., 2018). The analysis received roughly 427k tweets to train the proposed models of machine learning (ML). Sarcastic tweets accounted for 504 of the 5250 total, with the remaining 4746 tweets classified as non-sarcastic. The analysis results indicated the primary purpose of the research study, which was to enhance the performance of present sarcasm detection algorithms for Hindi-English language combinations that operate on code-mixed data classification. This goal was achieved because a data balancing layer was incorporated in the algorithm before classifying the extracted features.

KEYWORDS: Sarcasm, Natural Language Processing, Social Media Platforms, Data balancing layer, Machine Learning

1. INTRODUCTION

Nowadays, a significant proportion of human interaction occurs through social media. As a result, massive amounts of textual data are generated, enabling the drawing of crucial judgments. People from various cultures use SMPs like Facebook, Twitter, Reddit, and others to communicate and share their ideas and thoughts.

Due to the large amount of information available from SMPs, the study of various language expressions such as humor, irony, satire, hostility, enmity, and other types of sarcasm (Muresan et al., 2016) has emerged as a booming research area. Automatic detection of specific expressions is a challenging problem, primarily in the domain of NLP (Chowdhary, 2020). Automated detection and computational methods are used to determine if a specific emotion is there.

This research focuses on discovering one of the most commonly used language constructions throughout SMPs, *“sarcasm”*. Sarcasm, according to the Cambridge Dictionary (Bagate and Suguna, 2019), is *“the use of statements that manifestly suggest the reverse of what they represent”*. For instance, *“You have been working tirelessly”*, he noted to the blank page with a heavy touch of sarcasm.

2. LITERATURE REVIEW

Despite the fact that English is the most often used language on these websites, the vast majority of visitors are not native users of the English language. Because of this, some people choose to avoid communicating in English altogether. It has been discovered that nearly half of all tweets are written in a language other than English, according to the research of the most commonly used languages on Twitter for information sharing (De Cock and Pedraza, 2018). This method can now be used to manage multilingual data from social networking sites. According to several studies, approximately 26% of the Indian population is multilingual (Wilson, Hurst, and Wigglesworth, 2018). As a result, code flipping and code mixing are frequent.

Code mixing is the phrase used to describe when people in a group use more than one language at a sentence level (Shanmugalingam, Sumathipala, and Premachandra, 2018). People who are multilingual, particularly on social media, make use of a variety of languages to connect with one another. Code mixed data presents several challenges, including the large number of new constructions that result from combining the vocabulary and syntax of two distinct languages, the availability of only a small amount of annotated data, and the use of approaches that differ significantly from those used with monolingual data.

While much research has been done on detecting sarcasm in English (Davidov, Tsur, and Rappoport, 2010; Bamman and Smith, 2015), the identification of sarcasm in code-mixed languages such as Hinglish (Hindi-English) remains relatively unexplored. A random forest model is used to recommend the present condition performance on a dataset of 5000 Hinglish tweets (Swami et al., 2018).

Imbalanced classification entails classifying prediction models for datasets with significant class imbalance (Yap et al., 2014). There is a problem with dealing with imbalanced sets of data in that most ML algorithms tend to overlook and underperform on the minority class (MC), even though this class is generally the most relevant. Oversampling the minority population may aid in balancing out imbalanced sets of data (Huda et al., 2018). The most fundamental technique is to repeat samples from the minority class, although these repeated instances provide no more information to the model. However, multiple cases can be created by combining existing ones. The Synthetic Minority Oversampling Technique (SMOTE) is a way to add data to data that doesn't include enough of a certain group of people (Iosifidis and Ntoutsi, 2018).

SMOTE commences by identifying a random MC member and finding the k-nearest MC neighbors. The synthetic instance is then created by selecting one of the k-nearest neighbors at random as well as linking to it to form a line segment in the feature space. Synthetic cases are created by convexly integrating the 2 different cases. As a result, the method is effective at creating interesting new synthetic cases of the MC which are roughly similar in feature space to current MC examples (Elreedy and Atiya, 2019).

Borderline SMOTE is a popular form of SMOTE that includes collecting samples of the minority class that are misclassified, for example, using a k-nearest neighbor classification technique (Wanget al., 2017). Only the most difficult circumstances may be oversampled, resulting in increased resolution in these cases. Researchers Maciejewski and Stefanowski (2011) present a variant of the method wherein the MC is also oversampled for cases in which borderline appearances in the MC are misclassified. We believe that the Borderline-SMOTE technique will generate new synthetic examples only along the decision boundary of 2 classes, instead of supplying new synthetic cases for the MC at random.

Bedi et al., 2021 made two key contributions:

(1) They came up with the MaSaC set of data to help them figure out and classify humor and sarcasm in conversational language.

(2) They developed MSH-COMICS, a novel attention-rich neural architecture for classifying utterances.

They did significant studies on both tasks, modifying the multi-modal inputs and numerous MSH-COMICS sub-modules. In addition, we conduct comparative analyses of various methodologies now in use. They looked at the model and did a very detailed analysis of the results to figure out what it was good at and what it was bad at.

Kamath et al., 2021 examined and classified many techniques for detecting sarcasm, particularly on social media platforms. They discussed the difficulties associated with recognizing sarcasm as well as presented a particular application for detecting sarcasm in mixed Hindi-English tweets.

3. METHODOLOGY

3.1 Creating a Dataset

As per Swami et al. (2018), sarcastic tweets constitute 504 of the 5250 total, with the other 4746 tweets being classed as non-sarcastic. Due to the dataset's greater asymmetry as well as inadequate size, all ML models appeared to predict incorrectly that all tweets were not sarcastic. To satisfy the model's requirements, the researcher used a larger class-balanced dataset that these researchers proposed for this study.

3.2 Dataset Analysis

Swami et al. (2018) analyzed a set of data of 5250 English-Hindi code-mixed tweets, 504 of which were classified as sarcastic or ironic. The dataset includes two distinct categories of tweets:

(1) Sarcastic tweets that don't have the hashtags #sarcasm or #irony are considered to be sarcastic.

(2) Sarcastic tweets that do not include the hashtag #sarcasm. The search terms (hashtags) that were applied to scrape the tweets were used to build the annotation system.

The suggested technique assigned a positive sarcasm label to all cases returned with hashtags like sarcasm, irony, and so on, while assigning a negative sarcasm label to all examples returned with generic hashtags like cricket, Bollywood, and so on. However, despite the fact that this annotation method was prone to noise, the researcher manually checked the data collected by Swami et al. (2018) and found that the number of noisy cases was very small.

Furthermore, noisy cases were required to demonstrate that the models adapted well to the different datasets acquired. A balancing layer within the set of data (Swami et al., 2018) was essential in ensuring that ML models learned the correct trends and were not biased towards a single class. Primarily, Hinglish data was used to train embedding, which was then complemented with English data.

3.2 Data Pre-processing

Due to the noisy aspect of social media data, substantial pre-processing is required. Swami et al. (2018) developed a system for producing a set of data that deleted all '#' symbols and all references to '@'.It also removed unique phrases, defined as having fewer than ten occurrences in the entire dataset, as well as search terms like cricket, sarcasm, and so on, to prevent ML models from being biased toward specific words when learning. URLs and punctuation marks have also been removed.

3.4 Features

3.4.1 Word N-Grams

The term "word n-gram" represents the absence or presence in a tweet of a continuous series of n-words or tokens. In past studies, it was established that word n-grams are excellent features for detecting sarcasm. This study looked at all n-grams with n values ranging from one to five. In order to limit the feature space, it only looked at n-grams with features that appeared at least ten times in the corpus (Swami et al., 2018).

3.4.2 Character N-Grams

The term "character n-gram" represents the absence or presence of a continuous series of n characters in a tweet. Character n-grams, as demonstrated in previous studies, are critical for detecting sarcasm. This study looked at all n-grams with 'n' between one and three. If all of these character n-grams are included, the size of the feature vectors (FV) will be greatly enhanced. As a result, it only looked at n-grams which occurred at least 8 times in the data (Swami et al., 2018).

3.4.3 Sarcasm Indicative Tokens (SIT)

This characteristic indicates whether or not sarcastic tokens are present. For each language label, this study updated prior reported methods for recognizing indicative hashtags as well as retrieving SIT. The formula used to assign a value to each token was as follows:

$$Score(token) = max_{label \in Sarcasm-Set} \frac{freq(token, sarcasm\_label)}{freq(token)}$$

where, Sarcasm-Set = {YES, NO}.

For the purposes of this study, those tokens with a value of 0.6 as well as at least 5 instances in the dataset were considered to be characteristics of a sarcastic indication. The analysis identified tokens that corresponded to each of the language tags and included them in the FV. The criterion values for both values as well as occurrences were determined via trial and error (Swami et al., 2018).

3.4.4 Emoticons

This feature shows whether or not a tweet includes different types of emoticons. Several studies have been conducted in which emoticons have been used as a characteristic for detecting sarcasm. The study looked at a collection of 27 emoticons as features in this scenario (Swami et al., 2018).

4. RESULTS AND DISCUSSION

4.1 Implementation

The implementation of the proposed system was focused on the concept of short text classification. As mentioned earlier, the dataset used by Swami et al. (2018) was a text Twitter that involved Hindi-English code-mixed data. The supervised classification algorithms such as Linear SVM, RBF-kernel, Random Forest, Decision Tree, Naive Bayes, etc., used the short text data to classify tweets based on the sarcasm involved to categorize them as sarcastic as well as non-sarcastic tweets.

The novel idea proposed in this research focused on implementing a new approach called *"the balancing layer"*. The purpose of introducing this layer in the algorithm was to balance the data before feeding it as an input to the classification model. The following is the flowchart of the proposed algorithm:

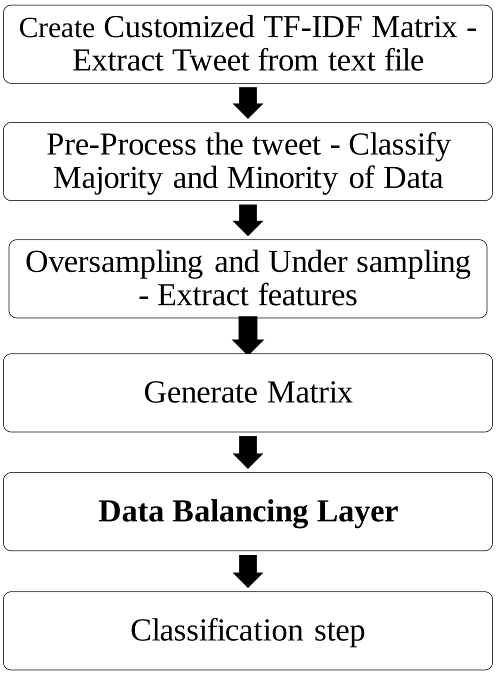
Create Customized TF-IDF Matrix - Extract Tweet from text file
Pre-Process the tweet - Classify Majority and Minority of Data
Oversampling and Under sampling - Extract features
Generate Matrix
Data Balancing Layer
Classification step

Figure 1: Proposed Algorithm

4.2 Working

The following steps explain the working of the proposed algorithm:

- The first step was to create a TF-IDF matrix (TM). The classification model used this matrix as an input to make its decisions. It was an extension of the BOW matrix, which was already in use in the current systems at the time of development. The TF-IDF scores of the retrieved features from the text are represented in the matrix. As an advancement to the same, the proposed system provided a TF-IDF score matrix for each extracted feature to enhance the performance of the existing classification model. This custom TF-IDF matrix was used for classifying the sarcastic and the non-sarcastic features in the tweets in an improved manner within the code-mixed data for the Hindi-English short text.

- The second step was to balance the data and divide it into positive and negative classes, commonly known as the minority and majority classes. The balancing layer was introduced at this step to increase the minority class of the sarcastic tweets and to similarly decrease the majority class of the non-sarcastic tweets. This implementation was carried out by the proposed algorithm in accordance with the existing systems to recommend the text suggestions based on the input. To implement this, the linear kernel algorithm was used instead of the cosine similarity algorithm to incorporate the similarity algorithm. This ensured performance improvement in terms of accuracy. Thus, a similarity matrix was formed by a cross product of the TF-IDF matrix.

TF-IDF matrix * TF-IDF matrix

- The third step was to utilize the similarity matrix created in the second step for oversampling the data. The term oversampling represents the process of identifying the tweets similar to highest-scoring minority features as in the matrix. In the proposed system, the Sampled TF-IDF matrix was created by taking the cross-product of the matrix with the highest scoring minority features. The process involved in this step was to attach the future and recommended sarcastic tweets to this TF-IDF matrix using a hybrid approach of the code-mixed data oversampling.

minority_class_TM * minority_class_TM

- The fourth step was to utilize the similarity matrix created in the second step for the under sampling of the majority class data. The term under sampling represents the process of balancing the dataset by eliminating the majority features and processing the minority class data. The proposed algorithm created a hybrid approach by forming a similarity matrix by the cross-product of the lowest TF-IDF score of the majority class feature and the minority class matrix. In this process, the non-sarcastic tweets from the majority class were deleted by the sampled_CTM matrix based on their similarity matrix to the sarcastic tweets in the minority class and majority class.

majority_class_CTM * minority_class_CTM

- The fifth and last step of the proposed algorithm was to input the data from the Sampled_CTM matrix to the classification model for detecting sarcasm prevailing in the Hindi-English code-mixed data to successfully implement this short text classification, which was the objective of the study.

4.3 Results Obtained

The F-score is defined as the mean of the accuracy and recall harmonics. Considering that the number of sarcastic tweets was fewer than the number of non-sarcastic tweets, the study attempted to utilize the F-score to examine the efficacy of the classification model, which was accomplished by the incorporation of a balancing layer into the current model. Due to this, it was determined that the system could not be evaluated just on the basis of its ability to detect errors.

Table 1: F-Scores for Existing Classifiers (Swami et al., 2018)

Features	Random Forest	RBF Kernel SVM	Linear SVM
Word n-grams	76.7	71.4	68.0
Character n-grams	75.0	73.1	66.4
Sarcasm Indicative Tokens	72.0	66.1	70.2
Emoticons	68.5	62.8	65.7
Average	78.4	76.5	71.7

As per Table 1, the current systems get an average F-score of 78.4 after performing tenfold cross validation using random forest classifier on the dataset. As shown in Table 1, each system received F-score for each feature individually and an overall F score for all features collectively. As can be seen, each strategy is influenced by a distinct attribute. For example, RBF kernel SVMs function better with character n-grams, random forest classifications with word n-grams, while linear SVMs perform better with SIT.

Table 2: F-Scores and ROC_AUC Scores for Proposed System

Classifiers	ROC_AUC Score			F-Score		
	SMOTE	**Borderline SMOTE**	**Proposed System**	**SMOTE**	**Borderline SMOTE**	**Proposed System**
Linear SVM	0.86	0.86	0.97	0.97	0.97	0.97
Random Forest	0.71	0.79	0.97	0.96	0.97	0.97
Decision Tree	0.91	0.92	0.97	0.97	0.97	0.97
RBF Kernel SVM	0.95	0.95	0.97	0.97	0.97	0.96

Table 2 shows the F-scores and ROC_AUC scores for the proposed system. Comparing the F-scores of the proposed system and that of the average scores of all the existing classifiers (Table 1; Swami et al., 2018), it is clear that the scores of the proposed system are quite higher. This increase in the F-scores is prevalent in all the classifiers and this level of increment can be attributed to the newly introduced balancing layer in the proposed system.

Similar to the F-score, the ROC_AUC score is used as an accuracy parameter to justify the results. ROC is an abbreviation for receiver operative characteristics. The curve is an assessment matrix that depicts the ratio of true positives to false positives in binary classification. Although AUC (area under the curve) is a metric of a classifier's ability to differentiate between different classes, the greater the value of the ROC AUC score, the better the classification performance is expected to be. It is also observed that the accuracy of the various classifiers in the proposed model is relatively high compared to the existing models. Whereas the average accuracy of the existing models is relatively low, it is clear that the accuracy is more than 90% for all the classifiers in the proposed system algorithm.

These findings are consistent with the significant purpose of the research, which is to enhance the performance of present sarcasm detection algorithms for Hindi-English language combinations using code-mixed data classification. Implementation of data balancing layer in the algorithm before classifying the extracted characteristics helps in attaining this objective successfully.

5. CONCLUSION

This study was able to provide a well-balanced Hindi-English code-mixed set of data for the detection of sarcasm by analyzing tweets from Twitter. It compared and contrasted the currently used and recommended algorithms, both of which were developed using data that was collected through scraping. For the sarcasm detection process, the researchers assessed the performance of several ML models that used newly formed word embeddings as input to address the problem. For the purpose of detecting sarcasm on Twitter, it showed the balancing layer contained within the English-Hindi code-mixed dataset. There was a detailed description of how sarcasm and language in tweets may be collected and analyzed at the tweet as well as token level.

It also demonstrated a baseline supervised classification method that was developed using the same set of data and using 3 different ML approaches as well as ten-fold cross validation.

The technique could be enhanced to evaluate vectors aligned with multilingual word embeddings made with MUSE to pre-aligned Fast Text word embeddings. BERT embeddings can also be analyzed and their performance on the similar task measured. For other language pairs which include other emotions, such as humor, comparable datasets can be produced. Additionally, this dataset can be standardized at the token level, which improves the performance of the classification system. Moreover, this set of data may be utilized to construct systems for automatic language recognition in code-mixed languages.

REFERENCES

[1] Bagate, R.A. and Suguna, R., 2019, September. Different Approaches in Sarcasm Detection: A Survey. In *International Conference on Intelligent Data Communication Technologies and Internet of Things* (pp. 425-433). Springer, Cham.

[2] Bamman, D. and Smith, N.A., 2015, April. Contextualized sarcasm detection on twitter. In *Ninth international AAAI conference on web and social media*.

[3] Bedi, M., Kumar, S., Akhtar, M.S. and Chakraborty, T., 2021. Multi-modal Sarcasm Detection and Humor Classification in Code-mixed Conversations. *IEEE Transactions on Affective Computing*.

[4] Chowdhary, K., 2020. Natural language processing. In *Fundamentals of artificial intelligence* (pp. 603-649). Springer, New Delhi.

[5] Davidov, D., Tsur, O. and Rappoport, A., 2010, July. Semi-supervised recognition of sarcasm in Twitter and Amazon. In *Proceedings of the fourteenth conference on computational natural language learning* (pp. 107-116).

[6] De Cock, B. and Pedraza, A.P., 2018. From expressing solidarity to mocking on Twitter: Pragmatic functions of hashtags starting with# jesuis across languages. *Language in society*, *47*(2), pp.197-217.

[7] Elreedy, D. and Atiya, A.F., 2019. A comprehensive analysis of synthetic minority oversampling technique (SMOTE) for handling class imbalance. *Information Sciences*, *505*, pp.32-64.

[8] Huda, S., Liu, K., Abdelrazek, M., Ibrahim, A., Alyahya, S., Al-Dossari, H. and Ahmad, S., 2018. An ensemble oversampling model for class imbalance problem in software defect prediction. *IEEE access*, *6*, pp.24184-24195.

[9] Iosifidis, V. and Ntoutsi, E., 2018. Dealing with bias via data augmentation in supervised learning scenarios. *Jo Bates Paul D. Clough Robert Jäschke*, *24*.

[10] Kamath, A., Guhekar, R., Makwana, M. and Dhage, S.N., 2021. Sarcasm Detection Approaches Survey. In *Advances in Computer, Communication and Computational Sciences* (pp. 593-609). Springer, Singapore.

[11] Maciejewski, T. and Stefanowski, J., 2011, April. Local neighbourhood extension of SMOTE for mining imbalanced data. In *2011 IEEE symposium on computational intelligence and data mining (CIDM)* (pp. 104-111). IEEE.

[12] Muresan, S., Gonzalez‐Ibanez, R., Ghosh, D. and Wacholder, N., 2016. Identification of nonliteral language in social media: A case study on sarcasm. *Journal of the Association for Information Science and Technology*, *67*(11), pp.2725-2737.

[13] Shanmugalingam, K., Sumathipala, S. and Premachandra, C., 2018, December. Word level language identification of code mixing text in social media using nlp. In *2018 3rd International Conference on Information Technology Research (ICITR)* (pp. 1-5). IEEE.

[14] Swami, S., Khandelwal, A., Singh, V., Akhtar, S.S. and Shrivastava, M., 2018. A corpus of english-hindi code-mixed tweets for sarcasm detection. *arXiv preprint arXiv:1805.11869*.

[15] Wang, Q., Xin, J., Wu, J. and Zheng, N., 2017, March. SVM classification of microaneurysms with imbalanced dataset based on borderline-SMOTE and data cleaning techniques. In *Ninth international conference on machine vision (ICMV 2016)* (Vol. 10341, pp. 355-361). SPIE.

[16] Wilson, A., Hurst, P. and Wigglesworth, G., 2018. Code-switching or code-mixing? Tiwi Children's Use of Language Resources in a Multilingual Environment. In *Language Practices of Indigenous Children and Youth* (pp. 119-145). Palgrave Macmillan, London.

[17] Yap, B.W., Rani, K.A., Rahman, H.A.A., Fong, S., Khairudin, Z. and Abdullah, N.N., 2014. An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets. In *Proceedings of the first international conference on advanced data and information engineering (DaEng-2013)* (pp. 13-22). Springer, Singapore.