# Performance analysis of token-based text augmentation techniques on text classification tasks in Indic languages

Rishabh Shirke
*Dept. of Information Technology*
*Indian Institute of Information Technology, Allahabad*
Prayagraj, India
mit2021098@iiita.ac.in

Anupam Agrawal
*Dept. of Information Technology*
*Indian Institute of Information Technology, Allahabad*
Prayagraj, India
anupam@iiita.ac.in

2023 9th International Conference on Smart Computing and Communications (ICSCC) | 979-8-3503-1409-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICSCC59169.2023.10335009

***Abstract*—Due to the increasing availability of large-scale datasets, deep learning models have become the first choice to tackle NLP tasks. However, training these models might require a huge amount of labeled data. It might be difficult to achieve great results if sufficient data is not available for training, which might be the case with low-resource languages. Fortunately, text augmentation has garnered considerable attention in recent years for improving the performance of natural language processing tasks, particularly in low-resource settings. The majority of these recently developed text data augmentation techniques have been thoroughly tested and analyzed in the resource-rich English language. However, there is a lack of similar performance analysis of text augmentation techniques in Indian languages. In this study, we focus on the performance of token-based text augmentation techniques for text classification tasks in Indian languages. We conduct performance analysis of token-based augmentation technique methods on headline and sentiment classification tasks in Indian languages using models like BiLSTM and the recently introduced ALBERT based multilingual model IndicBERT. We observed a significant performance boost across different languages and tasks with the BiLSTM model as compared to IndicBERT. IndicBERT, being pretrained on 12 Indian languages, has already learned semantic and contextual knowledge and thus does not benefit from added variations using augmentation.**

***Index Terms*—NLP, data augmentation, IndicBERT, headline classification, sentiment classification**

## I. Introduction

Advancements in the field of Natural language processing (NLP) have given rise to the creation of models that achieve good performance on various NLP tasks. However, these models are data-hungry and might not work as expected on unseen data if sufficient amount of labeled data is not available. To counter these problems and boost the performance of such models, data augmentation techniques can be used. Data augmentation techniques are used to generate new data using existing data. In the context of NLP, DA involves generating new data samples from existing samples without changing their meaning in order to preserve the labels. Although there has been a recent influx in the development of data augmentation techniques for various NLP tasks, the majority of studies are conducted on high-resource languages like English. As data augmentation is supposed to be more beneficial in a low-resource setting, a comprehensive study evaluating the performance of these augmentation techniques on low-resource languages can prove useful. Text classification is an important task in NLP, with applications ranging from spam detection to sentiment analysis. There has been a growing interest in developing text classification models for Indian languages, given the vast number of speakers and the importance of these languages in the global linguistic landscape. One challenge in developing text classification models for Indian languages is the availability of labeled data, which is often limited due to the complex and varied nature of the languages. To address the issue of the scarcity of labeled data, researchers have proposed various techniques for augmenting the available data, including token-based text augmentation methods. Token-based text augmentation involves manipulating the individual tokens (e.g., words or subwords) in a text to generate additional, semantically similar examples. These augmentation techniques have shown promising results. However, there have been limited empirical studies on the effectiveness of these techniques, specifically on text classification tasks in Indian languages. In this paper, we investigate the impact of token-based text augmentation techniques on text classification performance in Indian languages. We also investigate how the effectiveness of token-based text augmentation techniques varies with models like LSTM and IndicBERT model. Further, the paper is organized as follows: Section 2 gives an overview of related literature on data augmentation techniques and other similar efforts to evaluate the performance of augmentation techniques on languages other than English. Section 3 describes the token-based augmentation techniques and the models used. Section 4 describes the datasets used. In Section 5, we provide details of the experimental setup and the results of our experiments. In Section 6, we conclude by summarizing our findings and discussing the future scope.

## II. Related work

Survey by [1] lists data augmentation as a promising method to improve model's performance in low-resource settings. It

979-8-3503-1409-0/23/$31.00 © 2023 IEEE

also highlights the difference between the number of supported NLP tasks between high-resource and low-resource languages due to the lack of availability of quality-labeled and unlabeled data. [2] survey summarizes different text augmentation techniques and has briefly discussed directions for future research, including suggestions for analyzing the effectiveness of different augmentation techniques on low-resource languages and different language families in terms of linguistics and typological features. Token-level augmentation requires the replacement of a token with a similar token in order to maintain the semantics and preserve the label. [3] introduced EDA: Easy Data Augmentation, which includes simple token-level DA techniques like random swap, random deletion, synonym replacement, and random insertion for text classification tasks. The only external knowledge required for EDA is WordNet, which is used to get the synonyms. [4] proposed EDA inspired simple token-level DA based on swapping tokens with similar named entity tags for name entity recognition tasks. Instead of using an English thesaurus like WordNet, language models can be used to predict a similar word at a particular position using the context of the sentence. [5] used label-conditional bi-directional language models to predict words at a position without breaking the label compatibility. [6] used the conditional BERT model to generate semantically appropriate candidate words to be replaced with tokens. [7] carried out an in-depth comparative analysis of token and character level data augmentation techniques on a diverse set of languages other than English. [8] conducted an empirical study of EDA based text augmentation on Vietnamese social media texts.

## III. Methodology

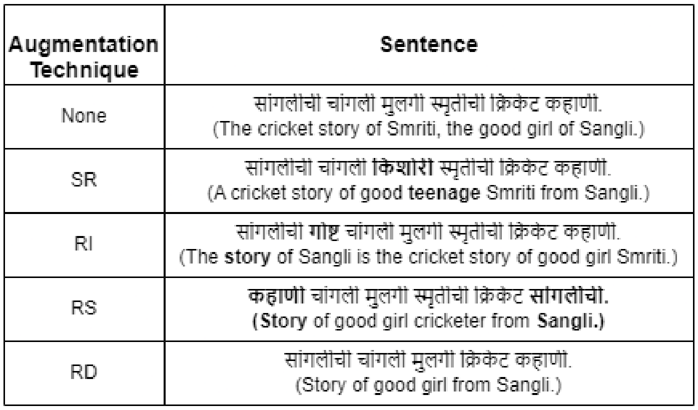
### A. *Augmentation methods used*

In this paper, we used EDA, a set of four easy token-based text augmentation techniques proposed by [3]. Following four augmentation techniques were proposed.

- **Synonym Replacement(SR)**: Replace $n$ random words with their synonyms. We used pyiwn library [9] to get the synonyms. Pyiwn is a Python-based API designed to facilitate access to Indian language WordNets.
- **Random Insertion(RI)**: Insert synonyms of $n$ randomly selected word in the sentence at random position.
- **Random Deletion(RD)**: Randomly remove each word in the sentence with probability $p$.
- **Random Swap(RS)**: Select two words from the sentence at random and swap their positions. Repeat this process $n$ times.

Where $n = \alpha l$, $l$ is the length of the given sentence and $\alpha$ is the parameter that controls the percentage of words changed in the sentences. For RD, $p = \alpha$. Figure 1 shows an example of EDA methods applied on a Marathi sentence.

### B. *The classification model*



| Augmentation Technique | Sentence |
|---|---|
| None | सांगलीची चांगली मुलगी स्मृतीची क्रिकेट कहाणी.<br>(The cricket story of Smriti, the good girl of Sangli.) |
| SR | सांगलीची चांगली **किशोरी** स्मृतीची क्रिकेट कहाणी.<br>(A cricket story of good **teenage** Smriti from Sangli.) |
| RI | सांगलीची **गोष्ट** चांगली मुलगी स्मृतीची क्रिकेट कहाणी.<br>(The **story** of Sangli is the cricket story of good girl Smriti.) |
| RS | **कहाणी** चांगली मुलगी स्मृतीची क्रिकेट **सांगलीची.**<br>(**Story** of good girl cricketer from **Sangli.)** |
| RD | सांगलीची चांगली मुलगी क्रिकेट कहाणी.<br>(Story of good girl from Sangli.) |

Fig. 1. Example showing different EDA augmentation techniques applied on a sentence in Marathi.

*1) BiLSTM:* For our experiments, we used LSTM-RNN architecture as described in [3]. The model is a two-layer bi-directional LSTM with hidden size of 64 in first layer and 32 in the second. Dropout layer with $p = 0.5$ is used after both LSTM layers. Linear layer of 20 hidden units with ReLU non-linearity followed by a softmax output layer. Categorical cross-entropy loss along with Adam optimizer was used for training the model.The method of early stopping was employed, where training is halted if there is no improvement in performance after 3 consecutive epochs. Fig. 2 shows the architecture of the BiLSTM model.

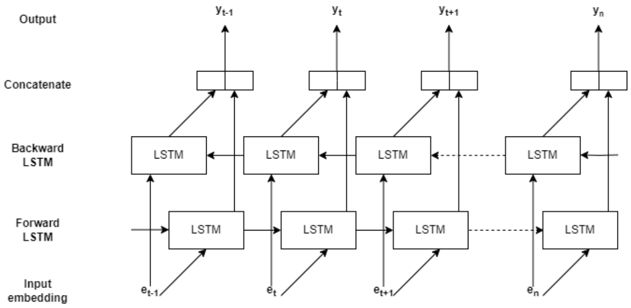


Fig. 2. BiLSTM architecture

*2) IndicBERT:* IndicBERT [10] is an ALBERT based multilingual NLU model pretrained on 12 Indian languages using masked language modeling objective. Backbone of the ALBERT model architecture is similar to BERT. Figure 3 shows the architecture of transformer encoder.

During fine-tuning we added linear classification head, which takes the representation of classification ([CLS]) token as input and generates a probability distribution over all potential categories. A learning rate $= 2e - 5$ along with a linear learning rate scheduler was used. We fine-tuned the model for five epochs with a batch size $= 16$.

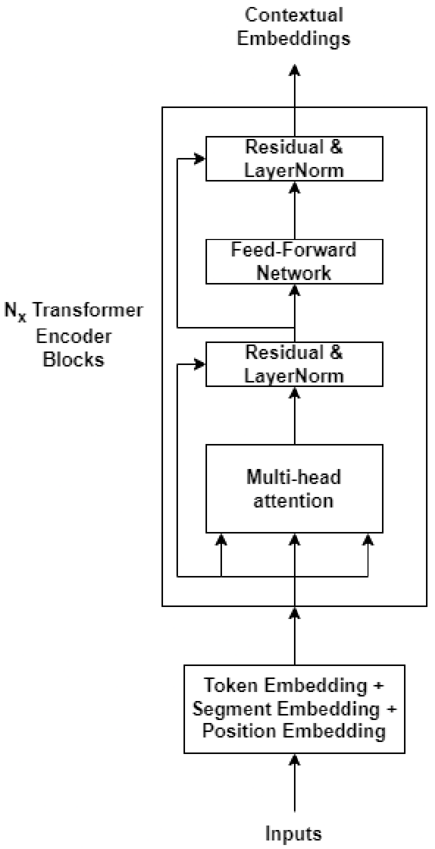


Fig. 3. Transformer encoder architecture

## IV. Datasets

*1) Headline classification:* For our experiments, we used publicly available Marathi, Gujarati, Tamil and Malayalam headline classification datasets provided by iNLTK [11]. Train, test, and validation split for each dataset is also provided by the author.

- iNLTK Marathi headline classification dataset(MR-HCD): Dataset consists of 12,092 headlines in the Marathi language. Each headline in the dataset is classified into three classes: Entertainment, state, and sports.
- iNLTK Gujarati headline classification dataset(GU-HCD): Dataset consists of 6,587 headlines classified into three classes: Entertainment, tech, and business.
- iNLTK Malayalam headline classification dataset(ML-HCD): Dataset has 6,296 Malayalam headlines classified into three classes: business, sports, and entertainment.
- iNLTK Tamil headline classification dataset(TA-HCD): Dataset has 5,346 Tamil headlines classified into three classes: business, Tamil cinema, and spirituality.

*2) Sentiment classification:*

- ACTSA Sentiment Classifcation(TE-Sent) [12]: Collection of 5407 Telugu sentences annotated as positive, negative, or neutral sentence.
- IIT-Patna Movie Reviews Dataset(HI-Sent) [13]: This dataset contains Hindi movie reviews categorized into three classes: neutral, negative, and positive.
- L3CubeMahaSent(MR-Sent) [14]: The dataset comprises 18,378 Marathi tweets categorized into three classes: neutral, negative, and positive.

Table I summarizes the split-wise statistics of the datasets.

TABLE I
Split-wise statistics of the datasets used

| Dataset | classes | Train size | Valid Size | Test Size |
|---|---|---|---|---|
| ML-HCD | 3 | 5036 | 630 | 630 |
| GU-HCD | 3 | 5269 | 659 | 659 |
| MR-HCD | 3 | 9672 | 1210 | 1210 |
| TA-HCD | 3 | 5346 | 669 | 669 |
| TE-Sent | 3 | 4327 | 540 | 540 |
| HI-Sent | 3 | 2479 | 309 | 309 |
| MR-Sent | 3 | 12114 | 1500 | 2250 |

## V. Results

### A. Experimental Setup

To investigate the performance of augmentation and how the performance is affected as training size changes, we experimented on the entire training dataset as well as on subset of size 500. The values used for $\alpha$ and $n_{aug}$ for different training sizes are given in Table II. $n_{aug}$ denotes the number of augmented sentences to generate from a single sentence and $\alpha$ denotes the percentage of words to be changed in the sentence. These values were selected by considering the suggestions given by [3]. As EDA comprises of four different augmentation techniques, we generate $n_{aug}/4$ sentences using each of these four techniques. A detailed explanation of the augmentation procedure is given in algorithm 1. Experiments are repeated five times with different random seed values.

**Algorithm 1:** Augmentation procedure

```
$D_{train} \leftarrow$ Training samples $\subseteq$ Entire training dataset
$n_{aug} \leftarrow$ augmented samples per training sample
$\alpha \leftarrow$ Fraction of words to be changed in a sentence
$k \leftarrow n_{aug}/4$
$D_{aug} \leftarrow []$
foreach sentence, label in $D_{train}$ do
    foreach transformation T in [SR, RI, RD, RS] do
        for i in range(k) do
            augmented $\leftarrow T(sentence,\alpha)$
            D$_{aug} \leftarrow D_{aug} \cup (augmented, label)$
        end
    end
end
```

### B. Augmentation performance results

For each dataset, we augmented the entire training set as well as a subset of size 500 and trained the BiLSTM and IndicBERT models on both the augmented as well as original data. The performance results of BiLSTM and IndicBERT models before and after augmentation on different training set

TABLE II
RECOMMENDED VALUE OF $n_{aug}$ AND $\alpha$ FOR DIFFERENT TRAINING SET SIZES

| $N_{train}$ | $n_{aug}$ | $\alpha$ |
|---|---|---|
| 500 | 16 | 0.05 |
| 2000 | 8 | 0.05 |
| 5000 | 4 | 0.1 |
| 5000+ | 4 | 0.1 |

sizes for headline and sentiment classification tasks are summarized in Table III and Table IV respectively. The accuracy values reported are the average of five experiments performed with different random seeds. The average performance change due to augmentation for headline and sentiment classification tasks across all languages for different training set sizes and models is summarized in Table V and Table VI respectively.

As seen in Table V and Table VI, on average, the BiLSTM model significantly benefits from augmentation, irrespective of the training size. Improvement in the model's performance after data augmentation can be attributed to added variations and noise to the training dataset which helps the model to generalize well. Compared to BiLSTM, IndicBERT got a minute improvement when $N_{train} = 500$ and little to no improvement when the entire training dataset is considered. Being a pretrained model, it has a good inherent understanding of the contextual relationships among words and phrases and was not able to benefit from added variations using augmentation.

TABLE III
PERFORMANCE OF BILSTM AND INDICBERT WITH AND WITHOUT AUGMENTATION ON HEADLINE CLASSIFICATION TASK

| | Dataset | $N_{train}$ | Accuracy(%) w/o EDA | Accuracy(%) w/ EDA |
|---|---|---|---|---|
| BiLSTM | ML-HCD | 500 | 68.57 | **84.25** |
| | | Full | 73.56 | **90.3** |
| | GU-HCD | 500 | 68.13 | **79.75** |
| | | Full | 77.87 | **86.98** |
| | MR-HCD | 500 | 70.38 | **78.41** |
| | | Full | 75.78 | **87.09** |
| | TA-HCD | 500 | 78.02 | **84.06** |
| | | Full | 84.06 | **91.00** |
| IndicBERT | ML-HCD | 500 | 76.98 | **80.00** |
| | | Full | 87.67 | **89.04** |
| | GU-HCD | 500 | 87.86 | **89.10** |
| | | Full | **90.89** | 90.59 |
| | MR-HCD | 500 | **81.48** | 80.57 |
| | | Full | 89.91 | **91.32** |
| | TA-HCD | 500 | 75.66 | **78.62** |
| | | Full | 86.24 | **86.84** |

TABLE IV
PERFORMANCE OF BILSTM AND INDICBERT WITH AND WITHOUT AUGMENTATION ON SENTIMENT CLASSIFICATION TASK

| | Dataset | $N_{train}$ | Accuracy(%) w/o EDA | Accuracy(%) w/ EDA |
|---|---|---|---|---|
| BiLSTM | HI-Sent | 500 | 41.61 | **46.70** |
| | | Full | 51.22 | **55.09** |
| | MR-Sent | 500 | 50.25 | **65.67** |
| | | Full | 71.94 | **77.38** |
| | TE-Sent | 500 | 46.21 | **44.87** |
| | | Full | 42.47 | **50.09** |
| IndicBERT | HI-Sent | 500 | 48.06 | **52.25** |
| | | Full | **56.77** | 56.45 |
| | MR-Sent | 500 | 64.42 | **68.46** |
| | | Full | 82.33 | **82.53** |
| | TE-Sent | 500 | **47.87** | 47.23 |
| | | Full | 58.96 | **59.14** |

TABLE V
AVERAGE PERFORMANCE CHANGE DUE TO AUGMENTATION FOR HEADLINE CLASSIFICATION TASK

| Model | $N_{train}$ | Average Accuracy(%) change |
|---|---|---|
| BiLSTM | 500 | 10.34 |
| | Full | 11.025 |
| IndicBERT | 500 | 1.57 |
| | Full | 0.77 |

## VI. CONCLUSION

In recent years, text augmentation has garnered substantial interest for enhancing the effectiveness of natural language processing tasks, especially in situations with limited available resources. The majority of these works used English language datasets to demonstrate the performance of the augmentation technique, which might not perform well on truly low-resource languages due to their differences in terms of linguistics and typological features. Although some efforts have been made to gauge the performance of data augmentation techniques on languages other than English, there has been a lack of such studies focusing on Indian languages. To address this situation, in this paper, we did a performance analysis of token-based augmentation techniques on text classification tasks in Indian languages and also investigated how the performance of deep learning models like BiLSTM and pretrained transformer models like IndicBERT vary with data augmentation. We observed that BiLSTM mode benefited the most across all the languages and classification tasks. IndicBERT being pre-trained on a huge corpus, was not able to benefit from augmentation. Further work can be done to analyze more advanced augmentation techniques on other Indian languages and NLP tasks, which could be very beneficial for NLP practitioners working on low-resource Indian languages.

## REFERENCES

[1] M. A. Hedderich, L. Lange, H. Adel, J. Strötgen, and D. Klakow, "A survey on recent approaches for natural language processing in low-resource scenarios," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, Jun. 2021, pp. 2545–2568. [Online]. Available: https://aclanthology.org/2021.naacl-main.201

[2] S. Y. Feng, V. Gangal, J. Wei, S. Chandar, S. Vosoughi, T. Mitamura, and E. Hovy, "A survey of data augmentation approaches for NLP," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Online: Association for Computational Linguistics,

TABLE VI
AVERAGE PERFORMANCE CHANGE DUE TO AUGMENTATION FOR SENTIMENT CLASSIFICATION TASK

| **Model** | $N_{train}$ | **Average Accuracy(%) change** |
|---|---|---|
| BiLSTM | 500 | 6.39 |
| | Full | 5.64 |
| IndicBERT | 500 | 2.53 |
| | Full | 0.02 |

Aug. 2021, pp. 968–988. [Online]. Available: https://aclanthology.org/2021.findings-acl.84

[3] J. Wei and K. Zou, "EDA: Easy data augmentation techniques for boosting performance on text classification tasks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 6382–6388. [Online]. Available: https://aclanthology.org/D19-1670

[4] X. Dai and H. Adel, "An analysis of simple data augmentation for named entity recognition," in *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020, pp. 3861–3867. [Online]. Available: https://aclanthology.org/2020.coling-main.343

[5] S. Kobayashi, "Contextual augmentation: Data augmentation by words with paradigmatic relations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 452–457. [Online]. Available: https://aclanthology.org/N18-2072

[6] X. Wu, S. Lv, L. Zang, J. Han, and S. Hu, "Conditional bert contextual augmentation," in *Computational Science – ICCS 2019*, J. M. F. Rodrigues, P. J. S. Cardoso, J. Monteiro, R. Lam, V. V. Krzhizhanovskaya, M. H. Lees, J. J. Dongarra, and P. M. Sloot, Eds. Cham: Springer International Publishing, 2019, pp. 84–95.

[7] G. G. Şahin, "To augment or not to augment? a comparative study on text augmentation techniques for low-resource NLP," *Computational Linguistics*, vol. 48, no. 1, pp. 5–42, Mar. 2022. [Online]. Available: https://aclanthology.org/2022.cl-1.2

[8] S. Luu, K. Nguyen, and N. Nguyen, "Empirical study of text augmentation on social media text in Vietnamese," in *Proceedings of the 34th Pacific Asia Conference on Language, Information and Computation*. Hanoi, Vietnam: Association for Computational Linguistics, Oct. 2020, pp. 462–470. [Online]. Available: https://aclanthology.org/2020.paclic-1.53

[9] R. Panjwani, D. Kanojia, and P. Bhattacharyya, "pyiwn: A python based API to access Indian language WordNets," in *Proceedings of the 9th Global Wordnet Conference*. Nanyang Technological University (NTU), Singapore: Global Wordnet Association, Jan. 2018, pp. 378–383. [Online]. Available: https://aclanthology.org/2018.gwc-1.47

[10] D. Kakwani, A. Kunchukuttan, S. Golla, G. N.C., A. Bhattacharyya, M. M. Khapra, and P. Kumar, "IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages," in *Findings of EMNLP*, 2020.

[11] G. Arora, "iNLTK: Natural language toolkit for indic languages," in *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 66–71. [Online]. Available: https://aclanthology.org/2020.nlposs-1.10

[12] S. S. Mukku and R. Mamidi, "ACTSA: Annotated corpus for Telugu sentiment analysis," in *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*. Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 54–58. [Online]. Available: https://aclanthology.org/W17-5408

[13] M. S. Akhtar, A. Kumar, A. Ekbal, and P. Bhattacharyya, "A hybrid deep learning architecture for sentiment analysis," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan: The COLING 2016 Organizing Committee, Dec. 2016, pp. 482–493. [Online]. Available: https://aclanthology.org/C16-1047

[14] A. Kulkarni, M. Mandhane, M. Likhitkar, G. Kshirsagar, and R. Joshi, "L3cubemahasent: A marathi tweet-based sentiment analysis dataset," in *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2021, pp. 213–220.