

ADA: An Attention-Based Data Augmentation Approach to Handle Imbalanced Textual Datasets

Amit Kumar Sah^[0000-0001-9218-5142] and
Muhammad Abulaish^[0000-0003-3387-4743]

Department of Computer Science, South Asian University, New Delhi, India
amitcsrs@students.sau.ac.in, abulaish@ieee.org

Abstract. This paper presents an Attention-based Data Augmentation (ADA) approach that extracts keywords from minority class data points using a vector similarity-based mechanism, uses the extracted keywords to extract significant contextual words from minority class documents using an attention mechanism, and uses the significant contextual words to enrich the minority class dataset. By creating new documents based on significant contextual words and adding them to the minority class dataset, we oversample the dataset for the minority class. On the classification job, we compare the original and oversampled versions of the datasets. We also compare ADA over the augmented datasets with two popular state-of-the-art text data augmentation methods. According to the experimental findings, classification algorithms perform better when used to augmented datasets produced by any data augmentation technique than when applied to the datasets' original versions. Additionally, the classifiers trained over the augmented datasets generated by ADA are more effective than those generated by state-of-the-art data augmentation techniques.

Keywords: Data Augmentation · Machine Learning · Deep Learning · Class Imbalance · Attention Mechanism · Information Extraction

1 Introduction

Textual data typically experiences problems with class imbalance. For instance, the proportion of fake, hateful, and spam tweets to actual tweets is low. It takes a lot of work to gather textual training data because the distribution of the gathered data must match that of the original data's syntax, semantics, and pragmatics. One of the most common methods for gathering data is oversampling, which involves producing more documents or samples from the minority class or repeating some documents. The textual dataset is oversampled by the text data augmentation mechanism using a variety of techniques. These strategies include copying documents, changing words with synonyms, or creating new data points using deep learning models. Data augmentation is one of the most popular methods for enhancing model generalization in deep learning models that successfully lowers overfitting while training a neural network. In the field

of image processing, data augmentation techniques are successfully used [12, 6]. Since they have syntactic, semantic, and pragmatic properties, data augmentation techniques that are effective for image data cannot be applied to textual data. The use of a thesaurus, synonyms, and similarities based on certain algorithms are typically involved in textual data augmentation. Although data augmentation can aid in the training of more reliable models, it is difficult to create universal rules for language transformation due to the complexity of Natural Language Processing (NLP). As a result, the main difficulty in proposing a generalized text data augmentation approach is NLP’s complexity. Keywords and keyphrases are crucial for text data augmentation, according to [1]. The keywords and keyphrases in a document serve to summarize its main points. One of the NLP-related issues with the most research is keyword extraction. Several methods that are frequently used for keyword extraction are – (i) Statistical methods, which primarily use term frequency and word distribution-based methods; (ii) Machine learning and deep learning-based methods, which employ a variety of supervised, semi-supervised, or unsupervised learning algorithms for keywords extraction; and (iii) Graph-based methods, which typically model the document’s vocabulary as nodes and connect them based on the relationships between the words.

In this paper, we present an Attention-based Data Augmentation (ADA) approach to oversample the minority class instances of imbalanced textual datasets to improve the detection efficacy of the classification algorithms. The proposed approach utilizes a vector similarity-based keywords extraction mechanism to identify keywords from the minority class data points. Using an attention mechanism, it exploits the identified keywords to extract its corresponding significant contextual words from minority class documents. Finally, it utilizes those significant contextual words to enrich the minority class dataset. The proposed approach oversamples the minority class dataset by generating new documents based on keywords and their significant contextual words and augmenting them to the minority class dataset. The proposed approach seems interpretable and improves the performance of the deep learning classifier over the augmented datasets.

In order to increase the detection accuracy of the classification algorithms, we describe in this research an Attention-based Data Augmentation (ADA) method to oversample the minority class instances in imbalanced textual datasets. The suggested method extracts keywords from the minority class data points using a process based on vector similarity. It uses an attention mechanism to extract significant contextual words from minority class documents that correlate to the discovered keywords. Finally, it makes use of those important contextual words to enhance the dataset for the minority class. By creating additional documents based on keywords and their significant contextual terms and adding them to the minority class dataset, the suggested approach oversamples the minority class dataset. The deep learning classifier performs better on the augmented datasets generated by the proposed technique, which is reportedly interpretable.

The remainder of the paper is structured as follows. An overview of the available text data augmentation literature is provided in section 2. The proposed

attention-based text data augmentation approach is fully described in section 3. The experimental setup and evaluation results are presented in section 4. Finally, the work is concluded with suggestions for future research in section 5.

2 Related Works

In the case of short text documents like reviews and tweets, where multiple words appear exceedingly seldom, data augmentation becomes crucial. In these circumstances, data augmentation becomes essential for deep learning models to increase their capacity for generalization. Researchers have made a contribution in this area by suggesting several text data augmentation strategies. In [11], authors performed data augmentation using English thesaurus and evaluated using deep learning models. In [8], the authors proposed to append original training sentences with their corresponding predicate-arguments triplets generated by a semantic role labeling tagger. In [10], authors introduced Easy Data Augmentation (EDA), showing that data augmentation using simple operations like synonym replacement, random insertion, random swap, and random deletion over a textual dataset can boost the performance of a classifier on text classification tasks. In [5], authors proposed contextual augmentation for labeled sentences by offering a wide range of substitute words, which a label-conditional bidirectional language model predicts according to the context. In [7], the authors explained that many data augmentation methods could not achieve gains when using large pre-trained language models because they are already invariant to various transformations. Instead, creating new linguistic patterns could be helpful. In [1], authors showed that augmenting n -grams from a minority class document that contains keywords extracted from a minority class dataset using Latent Dirichlet Allocation (LDA) to the same document can improve the performance of the CNN on textual datasets.

3 Proposed Data Augmentation Approach

This section discusses the proposed attention-based text data augmentation mechanism to handle imbalanced textual data. Table 1 gives the statistics of the Amazon reviews datasets used in our experiment. It can be observed from Table 1 that the ratio of the number of positive reviews to negative reviews, i.e., imbalance ratio (IR), is significantly high for all the datasets. So, we consider the positive reviews dataset as the majority class dataset (X_{maj}) and the negative reviews dataset as the minority class dataset (X_{min}). The main goal is to balance the dataset by augmenting the minority class with non-duplicate documents that incorporate additional knowledge to the minority class. In this process, we first extract keywords from the minority class based on a naive similar semantic space concept as discussed in section 3.1. After that, we create a keyword-based labeled dataset as discussed in section 3.2. We then deploy 2 parallel attention-based BiLSTM on the keyword-based labeled dataset to learn significant words belonging to each document of the minority class that contains the keyword(s) as

discussed in section 3.3. We then select the documents from the keywords-based dataset generated and labeled corresponding to keywords and transform them using a language model as discussed in section 3.4. Finally, we oversample the minority class dataset by augmenting the transformed version of the documents, as discussed in section 3.5. Figure 1 illustrates the workflow of the proposed approach.

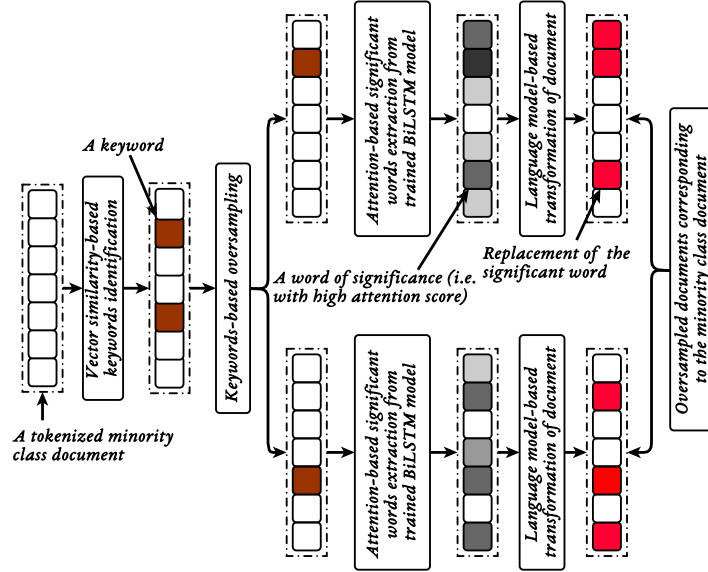


Fig. 1. Workflow of the proposed data augmentation approach

3.1 Vector Similarity-Based Keywords Extraction

In this section, we discuss how we extract keywords from the minority class (X_{min}) using “Bidirectional Encoder Representations from Transformers” (BERT). BERT is a bi-directional transformer model that helps to capture the meaning of words, phrases, and documents by encoding them to vectors. There is a general notion that word embedding of semantically similar words is close in vector space. With this notion, we propose identifying the keywords from the minority class dataset as those words whose word embedding representation is closer to that of the entire minority class dataset. For this, we first generate embedding corresponding to the entire minority class dataset, then generate the embeddings corresponding to each word in the vocabulary of the minority class dataset. Towards this direction, for document-level embedding, we prefer to use SBERT, a modification of the pre-trained BERT network originally presented in [9]. SBERT uses siamese

and triplet network structures and has proven to be a successful bi-encoder model for generating semantically meaningful sentence embeddings, which we can utilize for textual similarity comparisons using cosine similarity. SBERT generates semantically more acceptable and expressive sentence embeddings by fine-tuning the pre-trained BERT network.

At first, we encode the entire minority class documents to a single vector (V_{min}) using SBERT. We encode each document in X_{min} using SBERT to extract individual sentence-level embedding. We then average the sentence embeddings corresponding to all the documents in X_{min} to get a single minority class embedding vector V_{min} . After that, we encode i^{th} word from the minority class vocabulary ($Vocab_{min}$) to its corresponding embedding vector, w_i , using BERT. Finally, we calculate cosine similarity ($CoSimValue$) between embedding vector of each i^{th} word $\in Vocab_{min}$, w_i with V_{min} , to give $CoSimValue(w_i, V_{min})$ as given by equation 1, where $CoSimValue(w_i, V_{min}) \in [-1, 1]$.

$$CoSimValue(w_i, V_{min}) = \frac{w_i \cdot V_{min}}{\|w_i\| \|V_{min}\|} \quad (1)$$

We sort words in order of descending $CoSimValue$, and to balance the total number of review documents in both the classes in the dataset; we select only the top k words, as discussed in section 3.2. We refer to the top k words from $Vocab_{min}$ as the minority class keywords (K_{min}).

3.2 Keywords-Based Labeled Dataset Creation

In this section, we discuss the creation of binary labeled dataset D_{swe} for significant words extraction from minority class dataset X_{min} . Towards this direction, for each keyword $kw \in K_{min}$, in order of decreasing $CoSimValue$, we oversample each review document $r \in X_{min}$ with respect to every word $w \in r$. We assign class label 0 to the oversampled review if $w = kw$, and class label 1 otherwise. The main aim of creating this dataset is to generate additional minority class documents required to balance the dataset and extract the significant words of each review document labeled 0 as discussed in the upcoming section 3.3. So, we represent this dataset as significant words extraction dataset, denoted by D_{swe} , class 0 documents by D_{swe}^k dataset and class 1 documents by D_{swe}^{nk} dataset. We continue this process until the total number of documents in the minority class dataset, and significant words extraction dataset combined is equal to the number of documents in the majority class dataset, i.e., $|X_{min}| + |D_{swe}^k| = |X_{maj}|$.

3.3 Attention-Based Significant Words Extraction

In this section, we discuss the process of significant words identification from each minority class review document that contains the keyword(s). We identify the word corresponding to which a review document $r \in D_{swe}$ has been generated, as discussed in section 3.2, by target word w_t where $w_t \in r$. We aim to identify the words $w \in r$ that contributes the most when predicting the target word w_t using

the attention mechanism, which is well known for its ability to rank features. Here, we apply the attention mechanism to capture the informative parts of associated contexts. In order to achieve this, we pass each review document $r \in D_{swe}$ dataset through a pair of parallel attention-based 2-layers stacked BiLSTM, followed by a dense layer, and finally through a softmax layer.

Let us suppose r_i is the i^{th} review document, and $r_i \in D_{swe}$ such that $r_i = \{w_1, w_2, \dots, w_t, \dots, w_{n-1}, w_n\}$ where w_t is the target word and n is the number of words in the review document. Our model aims to learn the importance of each word $w \in r_i$ while training the model on r_i with emphasis on w_t , where w_t is a target word corresponding to which $r_i \in D_{swe}$ has been generated and labeled, as discussed in section 3.3. To this end, we have two parallel attention-based 2 layers stacked BiLSTM, one encoding the document from the beginning to the target word ($BiLSTM_b$), and the other from the target word to the end of the document ($BiLSTM_e$) given by equations 2 and 3 respectively.

$$h_{w_t}^b = BiLSTM_b(w_t, h_{w_{t-1}}^b) \quad (2)$$

$$h_{w_t}^e = BiLSTM_e(w_t, h_{w_{t-1}}^e) \quad (3)$$

where $BiLSTM_b$ and $BiLSTM_e$ are two employed BiLSTM that model the preceding and following context of the target word independently.

Not every word encoded by $BiLSTM_b$ and $BiLSTM_e$ are equally significant. In order to identify the more significant words, we have an attention layer at the top of $BiLSTM_b$ and $BiLSTM_e$, which helps decode the more significant/informative words by assigning them attention scores. We use the attention mechanism to assign a variable weight to all words (i) from the beginning of the review document to the target word (encoded by $BiLSTM_b$) and (ii) from the target words towards the end of the review (encoded by $BiLSTM_e$), depending on their contextual importance. For example, for encoded vector V_{r_i} corresponding to review document $r_i \in D_{swe}$; if hidden state representation of a target word $w_t \in V_{r_i}$ given by $BiLSTM$ is h_{w_t} , then it is passed to a dense-layer to learn its hidden representation h'_{w_t} , as given by equation 4, where W and B represent the weight and bias, respectively. Thereafter, similarity is calculated between h_{w_t} and a vertex vector v_{w_t} which represents the importance of $w_t \in V_{r_i}$. We also compute the normalized importance score of w_t using equation 5. The feature-level context vector v_{w_t} is randomly initialized and jointly learned during the training process. Finally, the attention-aware representation of the review document r_i is learned and represented as \mathcal{A}_{r_i} . It is computed as a weighted sum of the hidden representation of each word, as given by equation 6.

$$h'_{w_t} = \tanh(W h_{w_t} + B) \quad (4)$$

$$\alpha_{w_t} = \frac{\exp(h'_{w_t} v_{w_t})}{\sum_w \exp(h'_{w_t} v_{w_t})} \quad (5)$$

$$\mathcal{A}_{r_i} = \sum_w \alpha_{w_t} h_{w_t} \quad (6)$$

Both $BiLSTM_b$ and $BiLSTM_e$ goes through processes in equations 4, 5, and 6 simultaneously. As a result, the attention-based representation corresponding to $BiLSTM_b$ and $BiLSTM_e$ for review document r_i are obtained, represented as $\mathcal{A}_{r_i}^b$ and $\mathcal{A}_{r_i}^e$. Afterward, we concatenate these two vectors to generate the final representation vector of the review document r_i , pass it through a dense layer with 1024 neurons, and finally through a softmax layer with 2 neurons. We do this to make the model learn and identify the target word given the attention-based weight distribution of the contextual words.

We train the parallel attention-based $BiLSTM$ model on D_{swe} dataset. Once we have trained the model, we extract the attention-based vectors \mathcal{A}^b and \mathcal{A}^e . These vectors are the attention scores corresponding to words on both sides of the target word w_t . We rank the top words on both sides of w_t based on their attention scores. In this work, we've selected top 15% words corresponding to both the $BiLSTM_b$ and $BiLSTM_e$.

3.4 Language Model-Based Transformation of Review Documents

In this section, we discuss the process of language model-based transformation of review documents in D_{swe}^k . We aim to transform a review document $r \in D_{swe}^k$ to r_t such that the transformed review document r_t is a semantically similar but non-duplicate version of r . Towards this, we ensure that the words replaced from r to give r_t are contextually similar and have the semantically similar meaning as r . To this end, we deploy *Fill-Mask* task supported by *BERT*, where some of the words in a sentence are masked, and the *BERT* model predicts which words best replaces the current word, also known as *mask language modeling*. These models are helpful when we want to get a statistical understanding of the language in which the model is trained. As *BERT* is one of the best language models to date for this task, we prefer to use it for our work.

We have extracted top k significant words $S_w = \{S_{w_1}, S_{w_2}, \dots, S_{w_k}\}$ from each review document $r \in D_{swe}^k$, based on attention score as discussed in section 3.3. Now, for each i^{th} significant word $S_{w_i} \in S_w$, we replace it with its most similar word learned by masking and passing it through the *BERT* model. We follow the hold and predict strategy in which we mask one word and predict the words based on the rest words in the document. In this case, we mask words in order of importance, i.e., their attention score; when we mask a word, the rest of the words remain unchanged. The *BERT* model then gives the best word replacement for S_{w_i} in the form of $S_{w_i}^r$. We then replace S_{w_i} by $S_{w_i}^r$ and repeat this process for all the significant words in the review document r , in decreasing order of importance or attention score. Finally, we have the transformed review document r_t where all the words $w \in S_w \cap r$ are replaced by their best contextual and semantically similar words given by the *BERT* model.

3.5 Oversampling Minority Class Dataset

In this section, we discuss the oversampling process of the minority class dataset X_{min} . We first transform each review document r from the keywords-based

dataset D_{swe}^k to give the transformed review document r_t as discussed in section 3.4. As we know, ADA aims to balance the number of review documents in both classes of the review dataset. In section 3.2, D_{swe}^k has been created such that augmenting it to X_{min} gives the balanced dataset. Therefore, we augment D_{swe}^k with X_{min} to give oversampled minority class dataset AX_{min} , such that $|X_{maj}| = |AX_{min}|$. So, AX_{min} is the final augmented minority class dataset. We replace X_{min} by AX_{min} to give the oversampled balanced dataset.

4 Experimental Setup and Results

In this section, we present our experimental setup and discuss the evaluation of the proposed approach. We mention that experiments were performed on a machine with a 2.10 GHz Intel(R) Silver(R) processor and 192G RAM. Our attention-based text data augmentation model was implemented in Keras¹. For BERT pre-trained models, we used Transformers² library.

4.1 Datasets

We evaluate ADA over 3 publicly available Amazon reviews datasets [4], namely *musical instruments* (DS_1), *patio lawn and garden* (DS_2), and *automotive* (DS_3). We labeled all reviews with star ratings of 1 or 2 as negative reviews, whereas reviews with star ratings of 3, 4, or 5 as positive reviews. Table 1 presents the statistics of the modified datasets, listed in increasing order of the total number of reviews in the dataset. The IR value in Table 1 refers to the datasets’ imbalance ratio.

Table 1. Statistics of the Amazon review datasets

Dataset	#Reviews	$\#X_{maj}$	$\#X_{min}$	IR
DS_1	10,261	9,794	467	20.97
DS_2	13,272	12,080	1,192	10.13
DS_3	20,473	19,325	1,148	16.83

4.2 Data Preprocessing

The main issue with short text documents, especially review or tweet documents, is that they generally vary significantly from standard grammatical structures and possess predominantly creative spellings developed by the users due to character limitations and the habit of informal writing. Such data needs more special pre-processing than the standard pre-processing techniques, as we might face semantic loss. In order to avoid such a scenario, we performed the following pre-processing tasks: stop-words, URLs, and hashtag symbols removal, resolving elongated words, emoticons handling, resolving contractions, stemming, and lemmatization.

¹ <https://keras.io/>

² <https://huggingface.co/docs/transformers/index>

4.3 Classifier Architecture and Training Details

In this section, we present the classification technique used to validate the effectiveness of ADA. We used a 2-layer stacked BiLSTM architecture with 256 cells each, followed by the final softmax layer with 2 neurons, as we have formulated it as a binary classification problem. We have used Xavier Glorot initialization to assign initial weights, adam as an optimizer in our model. Our model used dropout at the BiLSTM and fully connected layers to minimize the overfitting effect, with probability values of 0.2 and 0.5, respectively. Further, our model used a $L2$ regularizer with a value of λ as 0.01 over the binary cross-entropy loss function. We used Rectified Linear Unit (ReLU) as an activation function throughout the model, except in the output layer, where we used the softmax function. We have used the softmax probability function in the last layer.

For classification tasks throughout this work, we have used 300-dimensional GloVe embeddings trained on the *Common Crawl* dataset with 840B tokens. For BERT-related tasks, we have used the *BERT base uncased* pre-trained model proposed in [2]. Table 2 gives the statistics of the total number of keywords extracted corresponding to different Amazon reviews datasets to generate the keyword-based labeled dataset, based on the discussion in section 3.2.

Table 2. Number of keywords extracted corresponding to different Amazon review datasets

Dataset	#Keywords
DS_1	2,076
DS_2	1,160
DS_3	2,494

4.4 Evaluation Metrics

There are very few metrics to consider when we require to evaluate the classifier on imbalanced data [3]. When the dataset is skewed, we should consider choosing evaluation metrics such that the classifier’s performance on the majority class does not overshadow its performance on the minority class. Hence, we evaluated the performance of the classification model throughout our experiments only for the minority class, and the macro averaged ones. We use precision (PR), recall (DR), F_1 measure (F_1), macro precision ($MacPR$), macro recall ($MacDR$), and macro F_1 ($MacF_1$) measure as evaluation metrics during the experimentations in this work. We chose these evaluation metrics to study the classifier’s performance on the minority class and observe whether there is any highly adverse impact on the majority class of the dataset.

4.5 Comparison Approaches

In order to establish the efficacy of the proposed model on imbalanced data, this section presents the comparative performance evaluation of ADA with the

Table 3. Comparative performance evaluation results of ADA on minority class

Approach	DS_1			DS_2			DS_3		
	PR	DR	F_1	PR	DR	F_1	PR	DR	F_1
Original Dataset	45.45	10.42	16.95	37.21	13.48	19.80	35.86	15.03	21.18
EDA[10]	95.37	98.92	97.11	90.86	97.90	94.25	90.35	98.69	94.33
CDA[5]	95.65	97.15	96.39	95.98	94.03	95.00	94.95	97.31	96.12
ADA	96.13	99.31	97.70	92.70	98.00	95.28	96.76	98.83	97.78

Table 4. Macro comparative performance evaluation results of ADA

Approach	DS_1			DS_2			DS_3		
	$MacPR$	$MacDR$	$MacF_1$	$MacPR$	$MacDR$	$MacF_1$	$MacPR$	$MacDR$	$MacF_1$
Original Dataset	70.61	54.90	57.25	64.61	55.62	57.30	65.48	56.71	58.95
EDA[10]	97.12	97.03	97.04	94.17	93.56	93.71	94.72	95.87	95.16
CDA[5]	96.44	96.46	96.45	95.13	95.09	95.10	96.15	96.15	96.16
ADA	97.64	97.28	97.43	95.30	95.17	95.16	97.50	96.99	97.23

following two standard text data augmentation techniques, namely – (i) **EDA – Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks** [10], and **Contextual Augmentation – Data Augmentation by Words with Paradigmatic Relations** [5].

4.6 Evaluation Results and Comparative Analysis

We oversampled the minority class of the original or non-augmented dataset by augmenting new review documents generated from the proposed attention-based text data augmentation technique to give a balanced dataset. We evaluated both the original and balanced versions of datasets on the BiLSTM model in order to study the effectiveness of our text data augmentation mechanism. Similarly, for the comparative study, works done in [10, 5] were evaluated following a similar approach, i.e., we oversampled the minority class dataset with new review documents generated from the respective text data augmentation mechanisms such that it resulted to a balanced dataset. We consider the evaluation metrics discussed in section 4.4 for evaluation purpose. We trained the BiLSTM model on 56% of the dataset, validated it on 14%, and finally tested the model on 30% unseen data. We trained the BiLSTM model for 100 epochs with early stopping as a regularization mechanism to combat overfitting and have recorded the results obtained on test data. Table 3 lists the classifier’s performance on the minority class dataset, and Table 4 lists the evaluation results of the classifier macro averaged over both the majority and the minority class dataset.

Performance on Minority Class: Table 3 shows that the DR value in particular, on the original datasets, was extremely poor and ranged between a minimum of 10.42% for DS_1 and 15.03% for DS_3 . However, we observed a radical

significant margin improvement on the oversampled datasets, with a minimum of 94.03% for DS_2 using CDA and a maximum of 99.31 for DS_1 using our proposed approach. ADA outperformed both the EDA and CDA in terms of DR and F_1 value over all the datasets. It was fascinating to observe that the DR value on the oversampled version of the datasets obtained using the proposed approach never fell below 98%, which was for the DS_2 dataset. We also observed that the F_1 value on the oversampled version of the datasets obtained using ADA never fell below 95.28% for DS_2 and reached the maximum 97.78% in the case of DS_3 . Further, in terms of PR , the performance of ADA was reported to be better than EDA and CDA on DS_1 and DS_3 , while CDA surpassed ADA on DS_2 . We observed that EDA surpassed CDA over all the datasets in terms of DR , whereas in terms of PR and F_1 , EDA surpassed CDA over all the datasets except DS_2 .

Macro Performance: Table 4 shows that ADA surpasses EDA and CDA in terms of all $MacPR$, $MacDR$, and $MacF_1$. Even in terms of $MacF_1$, ADA beat EDA and CDA by a wider margin on DS_3 , the largest dataset, than DS_1 and DS_2 , the other two smaller datasets. The reported macro-averaged performance result suggests that ADA generates a qualitative augmented and oversampled dataset, which remarkably improves the classifier performance on the minority class and does not hamper its performance on the majority class.

We observed that CDA performed comparatively better than EDA over DS_2 and DS_3 datasets on all evaluation criteria except DR . However, in terms of recall performance, EDA is better than CDA and comparable to ADA. Also, the performance analysis in section 4.6 and 4.6 suggests that ADA gives the best text data augmentation model compared to EDA and CDA.

5 Conclusion and Future Work

An Attention-Based Data Augmentation (ADA) method is presented in this paper as a solution to the class imbalance issue in processing textual datasets. Compared to the state-of-the-art methods (EDA and CDA), ADA offers observable advantages. For deep learning models that extract patterns from the data, the oversampled augmented dataset may be useful. It appears to be extremely helpful for fields with tiny and unbalanced datasets because it attempts to solve the issue of information scarcity. It appears like a potential topic of research to investigate various keyword extraction processes and provide a unique model to learn the best phrases that can replace the significant words discovered in section 3.3. To process imbalanced textual datasets, we are attempting to improve our suggested approach to produce more qualitative documents.

References

- [1] Abulaish, M., Sah, A.K.: A text data augmentation approach for improving the performance of CNN. In: 11th Int'l Conference on COMSNET, Bangalore, India. pp. 625–630 (2019)

- [2] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the Conference of the North American Chapter of the ACL: Human Language Technologies, Minnesota. pp. 4171–4186 (2019)
- [3] Ferri, C., Hernández-Orallo, J., Modroiu, R.: An experimental comparison of performance measures for classification. *PR Letters* **30**(1), 27–38 (2009)
- [4] He, R., McAuley, J.: Ups and Downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In: Proceedings of the 25th Int’l Conference on WWW. p. 507–517 (2016)
- [5] Kobayashi, S.: Contextual augmentation: Data augmentation by words with paradigmatic relations. In: Proceedings of the Conference of the North American Chapter of the ACL-HLT, Louisiana. pp. 452–457 (2018)
- [6] Krizhevsky, A., Sutskever, I., Hinton, G.E.: In: F. Pereira et al. (ed.) *Advances in Neural Information Processing Systems*, Nevada, USA
- [7] McCoy, T., Pavlick, E., Linzen, T.: Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In: Proceedings of the 57th Annual Meeting of the ACL, Florence, Italy. pp. 3428–3448 (2019)
- [8] Min, J., McCoy, R.T., Das, D., Pitler, E., Linzen, T.: Syntactic data augmentation increases robustness to inference heuristics. In: Proceedings of the 58th Annual Meeting of the ACL. pp. 2339–2352 (2020)
- [9] Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using siamese BERT-networks. In: Proceedings of the Conference on EMNLP and IJCNLP, Hong Kong, China. pp. 3982–3992 (2019)
- [10] Wei, J., Zou, K.: EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In: Proceedings of the Conference on EMNLP and IJCNLP, Hong Kong, China. pp. 6382–6388 (2019)
- [11] Zhang, X., Zhao, J.J., LeCun, Y.: Character-level convolutional networks for text classification. In: *Advances in Neural Information Processing Systems*, Quebec, Canada. pp. 649–657 (2015)
- [12] Zhong, Z., Zheng, L., Kang, G., Li, S., Yang, Y.: Random erasing data augmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence, New York. vol. 34, pp. 13001–13008 (2020)