This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Improving Turkish Text Sentiment Classification through Task-Specific and Universal Transformations: An Ensemble Data Augmentation Approach

AYTUG ONAN[1] and KADRIYE FILIZ BALBAL[2]

[1]Department of Computer Engineering, Faculty of Engineering and Architecture, İzmir Katip Çelebi University, 35620, İzmir, Turkey
[2]Department of Computer Science, Faculty of Science, Dokuz Eylül University, 35160, İzmir, Turkey

Corresponding author: Aytuğ Onan (e-mail: aytug.onan@ikcu.edu.tr).

ABSTRACT The exponential growth of digital data in recent years has spurred a significant interest in natural language processing (NLP) and sentiment analysis. However, the effectiveness of NLP models heavily relies on the availability of large, annotated datasets, which are often scarce or entirely absent for numerous languages, including Turkish. This scarcity of annotated data for Turkish presents a formidable obstacle in developing NLP models for the language. To overcome this challenge, various techniques have been proposed to augment the size of annotated datasets, with text data augmentation emerging as a promising solution. Text data augmentation involves the generation of synthetic data by transforming existing data, thus expanding the diversity and volume of the annotated dataset. While this technique has shown remarkable success in bolstering the performance of NLP models, its exploration in the context of Turkish and other low-resource languages has been limited. This paper introduces a novel ensemble approach to text data augmentation tailored for Turkish text sentiment classification. Our approach integrates both task-specific and universal transformations, capitalizing on the strengths of each to enrich the training dataset. We evaluate our proposed approach on the TRSAv1 dataset and compare it with established data augmentation techniques. Our experimental results demonstrate that our ensemble method achieves superior accuracy in sentiment classification compared to conventional techniques. Additionally, we conduct an in-depth analysis to assess the impact of individual transformation functions on classification performance. Our contribution lies in bridging the gap in research on data augmentation techniques tailored to Turkish NLP, emphasizing the need for more advanced ensemble methods, and offering benchmarking results that pave the way for the development of precise NLP models not only for Turkish but also for other low-resource languages.

INDEX TERMS text classification, sentiment analysis, ensemble learning, deep learning

I. INTRODUCTION

The rapid growth in digital data in recent years has brought about a rising interest in natural language processing (NLP) and sentiment analysis. However, the performance of NLP models highly depends on the avail- ability of large annotated datasets, which are often scarce or unavailable for many languages, including Turkish. This scarcity of annotated data for Turkish poses a significant challenge for the development of NLP models for the language. To overcome this limitation, various techniques have been proposed in the literature to increase the size of annotated datasets, such as text data augmentation.

Text data augmentation is the process of generating synthetic data by transforming the original data, which helps to increase the size and diversity of the annotated dataset [1]. This technique has been widely used in many NLP tasks such as sentiment analysis, named entity recognition, and machine translation [1] [2]. Conventional techniques of text data augmentation in NLP include data sampling, data generation, data transformation, and data augmentation through pre-training [3]. Data sampling techniques include oversampling and under

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

sampling, which balance the distribution of classes in the dataset. Data generation techniques include bootstrapping, generating synthetic data using generative models, and data augmentation through back-translation. Data transformation techniques include random perturbation, synonym substitution, and lexical substitution [4]. Data augmentation through pre-training involves pre-training models on large amounts of unannotated data and then fine-tuning on the annotated dataset.

The shortage of annotated datasets is not limited to Turkish; it is a common problem for low-resource languages [5]. Low-resource languages are those with limited resources such as annotated datasets, tools, and models for NLP. As a result, the development of NLP models for low-resource languages is challenging, and the performance of these models is often lower compared to high-resource languages. This issue is particularly relevant for Turkish as it is a low-resource language with limited annotated datasets available for NLP tasks.

Collecting annotated datasets for low-resource languages such as Turkish presents significant challenges due to the limited availability of language resources such as lexicons, morphological analyzers, and named entity recognition tools [6]. Moreover, the complexity of the Turkish language, including its rich vocabulary, multiple dialects, and diverse writing styles, compounds the scarcity of annotated data for Turkish [7]. As a result, developing NLP models for Turkish is particularly challenging, and their performance is often inferior to that of high-resource languages. Therefore, there is a need for novel techniques to address these challenges by increasing the size and diversity of annotated datasets in Turkish, such as text data augmentation, to enhance the performance of NLP models for this language.

The difficulty of collecting annotated datasets for Turkish is compounded by the fact that Turkish is an agglutinative language, meaning that words are formed by combining different morphemes, making the parsing and annotation of text data more challenging. Moreover, Turkish is a rich and complex language, with a vast vocabulary, multiple dialects, and diverse writing styles, making the development of NLP models for the language even more challenging [8]. Despite these challenges, the demand for NLP models for Turkish is increasing, especially in applications such as sentiment analysis, machine translation, and text classification. Sentiment analysis, in particular, is a critical application of NLP, and the demand for this technology is growing rapidly. Sentiment analysis refers to the process of determining the sentiment or emotion expressed in a text, such as positive, negative, or neutral. The technology is widely used in a variety of applications, including marketing, customer service, and political analysis [9].

This study aims to address the challenge of collecting annotated datasets for Turkish by proposing a novel text data augmentation technique based on the ensemble of task-specific and universal transformations. The proposed technique generates synthetic data by transforming the original data using a combination of task-specific and universal transformations. The task-specific transformations are de- signed to preserve the sentiment of the original text, while the universal transformations are designed to increase the diversity of the generated data. The proposed ensemble text data augmentation technique is evaluated on a Turkish sentiment analysis task, and the results demonstrate the effectiveness of the proposed technique compared to the baseline techniques.

In this paper, we present a text data augmentation approach based on an ensemble of task-specific and universal transformations. Our approach combines the use of task-specific transformations, such as sentence order swap, extending adjustment, and double denial, with universal transformations, such as Swap Synonym-WordNet, masked language model based suggestion, and insert adverb. This approach is motivated by the idea that task-specific transformations are better suited to capture the characteristics of specific NLP tasks, while universal transformations can be applied more broadly to improve the performance of NLP models in a wider range of tasks. The scientific contribution of this article is the development and evaluation of a novel ensemble approach for text data augmentation to improve Turkish sentiment analysis. Specifically, the main contributions of this study are: Introducing a novel text data augmentation technique that combines task-specific and universal transformations to generate synthetic training examples for Turkish text sentiment classification. Proposing a set of task-specific transformations, such as sentence order swap, extending adjustment, and double denial, and universal transformations, such as Swap Synonym- WordNet, masked language model based suggestion, and insert adverb-adverb, to increase the diversity of the generated data. Conducting a comprehensive experimental evaluation of the proposed approach and comparing it with several baseline methods. Demonstrating the effectiveness of the proposed approach in improving the performance of NLP models com- pared to conventional text data augmentation approaches. The proposed approach can be used to increase the size and diversity of annotated datasets for Turkish sentiment analysis, thereby improving the performance of NLP models for this language. Additionally, the approach can be adapted and applied to other NLP tasks and languages to increase the diversity and size of annotated datasets.

The rapid growth in digital data in recent years has brought about a rising interest in natural language processing (NLP) and sentiment analysis. However, the performance of NLP models highly depends on the availability of large annotated datasets, which are often scarce or unavailable for many languages, including Turkish. This scarcity of annotated data for Turkish poses a significant challenge for the development of NLP models for the language. To overcome this limitation, various techniques have been proposed in the literature to increase the size of annotated datasets, such as text data augmentation. While ensemble methods in data augmentation are well-established in other domains, their exploration in the

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

context of NLP, especially for low-resource languages like Turkish, remains relatively uncharted territory. In this paper, we propose a novel ensemble approach to text data augmentation for Turkish sentiment analysis, where we combine both universal transformations and task-specific techniques. While our ensemble approach might not be as intricate as some ensemble methods used in other domains, it makes a crucial contribution by addressing the dearth of research on data augmentation techniques tailored to Turkish NLP. Our primary goal is to establish a foundation for future research in this domain, highlighting the need for more advanced ensemble techniques, and providing benchmarking results that contribute to the development of accurate NLP models for Turkish and other low-resource languages.

Despite the growing interest in sentiment analysis for various languages, Turkish text sentiment analysis remains challenging due to the limited availability of annotated datasets and the complex morphological structure of the Turkish language. This research addresses the need for effective data augmentation techniques tailored to the nuances of Turkish sentiment analysis. The primary objectives of this study are to develop an innovative ensemble data augmentation method specifically designed for Turkish sentiment analysis and to demonstrate its effectiveness in enhancing classification accuracy. This work contributes to the field by introducing a novel ensemble approach that integrates both task-specific and universal transformation functions, a first of its kind for Turkish sentiment analysis. We also provide extensive experimental results to validate the efficacy of our proposed method. The motivation for this research stems from the critical gap in effective NLP tools for Turkish sentiment analysis, a gap that hinders the extraction of valuable insights from Turkish textual data. Our work aims to bridge this gap by providing a robust solution to enhance data quality and quantity for Turkish sentiment analysis.

The rest of this paper is organized as follows. Section 2 reviews the related work on text data augmentation in sentiment analysis. Section 3 presents our proposed methodology for text data augmentation based on an ensemble of task-specific and universal transformations. This includes the introduction of the task-specific transformation, as well as the use of universal transformations such as synonym substitution and random insertion. Section 4 presents the experimental results and a discussion of the results obtained using our proposed approach on a Turkish text sentiment classification task. Finally, Section 5 concludes the paper and provides suggestions for future work.

II. RELATED WORK

Text data augmentation methods can be broadly divided into two categories, as feature space (i.e., noise, and interpolation) and data space (i.e., character-level, word-level, phrase-level, and document-level) methods [4]. Character- based methods include noise and rule-based models, word- level methods include noise, language models, embeddings, and synonyms, phase-based methods include structure, and interpolation, document-level methods include translation, and generative models [4]. The rest of this section briefly summarizes the earlier research contributions on text data augmentation.

Belinkov and Bisk [10] proposed a character-based method to make neural machine translation (NMT) models less susceptible to adversarial examples by adding artificial and natural noise to the training data. The method involved adding artificial noise through operations such as random letter switching, mid-part word randomization, complete word randomization, and replacement of a letter with a neighboring letter on the keyboard. They also added natural noise by using parallel data and back-translation. In their experiments, the proposed method improved the robustness of NMT models against both natural and adversarial noise. Similarly, Feng et al. [11] applied random deletion, swap, and insertion of characters to the prompt portion of texts that were fine-tuned on text generators. They ignored the first and last character of a word and evaluated their method based on diversity, fluency, semantic context preservation, and sentiment consistency. They demonstrated that their augmentation approach outperformed the baseline in all aspects and could be extended to text classification tasks.

According to Coulombe [12], a method for inducing weak textual sounds involved changing, deleting, and adding letters in words, as well as modifying upper and lower case and punctuation. The best performing baseline was improved by 2.5% in terms of absolute accuracy. Wei and Zou [13] presented a text data augmentation model, referred as EDA (easy data augmentation), that uses four simple but effective operations: synonym replacement, random insertion, random swap, and random deletion. The authors demonstrate the efficacy of EDA on five text classification tasks using both convolutional and recurrent neural networks.

Li et al. [14] proposed a combination of syntactic and semantic noise generation methods with word dropout for text data augmentation. The syntactic noise was produced by shortening sentences, altering adjectives, and relativizing modifiers. The semantic noise was generated by replacing words with their synonyms. In comparison, word dropout was a more directly comparable noise generation method, where random input neurons or words were masked out during training. Rizos et al. [15] presented a novel method for data augmentation in which instances were shifted within the padding limits to increase the variability of the training data. Their approach was evaluated on a hate speech detection dataset and resulted in an improvement of more than 5.8% (Macro-F1). In a similar vein, Sun and He [16] proposed a method in which meaningless words were added either at the beginning or at end of instances.

Kolomiyets et al. [17] were among the first to utilize word substitution for data augmentation. Their approach involved substituting temporal expressions with potential synonyms from WordNet. The authors argued that substituting a single token in a sentence would generally preserve its semantics. In the context of time expression recognition, they suggested replacing the headword, as temporal trigger words tend to appear there. In a similar way, Mosolova et al. [18] applied synonym substitution

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

for text data augmentation. To improve the classification of annoying tweets, Wang and Yang [19] employ augmentation through synonym substitution. They apply k-nearest-neighbor to identify the embeddings that best replace the words in the training data. Using logistic regression, they show that their method results in an improvement of up to 2.4% in the F1-Score, compared to the baseline. Feng et al. [20] proposed a method to change the meaning of a text while maintaining its fluency and sentiment. The Semantic Text Exchange method involves identifying phrases in the original text that can be replaced by a replacement entity based on their constituents. Then, phrases similar to the identified phrases are masked and replaced with the attention-based language model to better suit the replacement entity. Şahin and Steedman [21] proposed two text data augmentation techniques using dependency trees to improve the performance of low-resource languages in NLP tasks. The techniques involve cropping sentences by removing dependency links and rotating sentences by moving the tree fragments around the root. The techniques are applied to the training sets of low-resource languages in the Universal Dependencies project, and a character-level sequence- tagging model is used to evaluate their effectiveness on part- of-speech tagging tasks. The study shows that the crop and rotate techniques provide improvements over models trained with non-augmented data, especially for languages with rich case marking systems. In a recent study, Şahin [22] presented a comprehensive analysis of text data augmentation techniques on NLP tasks for low-resource languages. Three categories of augmentation methods are investigated, based on syntax, token, and character level changes. The methods are compared on part-of-speech tagging, dependency parsing, and semantic role labeling tasks for a diverse set of languages. Results show that augmentation most significantly improves de- pendency parsing, followed by part-of-speech tagging and semantic role labeling. The character-level methods are identified as the most consistent performers, while synonym replacement and syntactic augmenters provide inconsistent improvements.

Rahmani et al. [23] proposed a new method for multimodal sentiment prediction that considers cognitive cues in a user-specific manner. The authors addressed the limitation of existing approaches that disregard user-specific aspects and presented a cognitive-aware framework that constructs an adaptive tree by hierarchically dividing users. The framework utilizes Long short-term memory submodels with attention-based fusion to transfer cognitive-oriented knowledge within the tree and generates conclusive agglomerative knowledge from the adaptive tree to predict final sentiments. The proposed model surpasses trending baselines and can better utilize latent cues, benefitting from multiple categorization strategies. The empirical results on real-world datasets indicate that the proposed hierarchical categorization model can better group users within the adaptive tree. In another study, Najafipour et al. [24] proposed a neural network-based temporal-textual framework that generates subgraphs with highly correlated authors from short-text contents. The authors addressed the challenges of noisy, ambiguous, and temporally skewed short-text contents and the inability of traditional text mining methods to extract concepts through words and phrases. The proposed framework effectively overcomes these challenges by generating highly correlated subgraphs. Saaki et al. [25] developed a value-wise ConvNet for Transformer models to create an infinite time-aware recommender system. The authors addressed the challenge of noise in short text contents, mismatched words, and the reluctance dilemma of experts to answer input queries. The proposed embedding approach constructs context-aware vectors, extracts knowledge domains from the online contextual content, and tracks user textual-temporal behavioral patterns via an infinite continuous-time module. The proposed system recommends a set of experts pertinent to the given query. In another study, Kamran et al. [26] developed EmoDNN, a deep neural network ensemble to infer the cognitive aspect of individuals and extract emotions from short texts. The authors addressed the challenge of de- vising a cognition-aware algorithm to trace emotional cues in brief contents. The proposed ensemble method aggregates outcomes from various classifiers to extract emotions from short texts and significantly outperforms existing methods. As the earlier research contributions indicate text data augmentation can play a crucial role in advancing the state- of-the-art for low-resource languages, which are often underrepresented in NLP research and lack sufficient annotated data.

While our paper primarily focuses on an ensemble text data augmentation approach that combines both task-specific and universal transformations, we acknowledge that contextual augmentation is an important area of research in NLP. Recent advances in contextual augmentation, including techniques that leverage large-scale pre-trained language models like BERT, GPT-3, and RoBERTa, have shown promising results in various NLP tasks. These models have the capability to generate contextually relevant and coherent text, making them valuable for data augmentation purposes. Some of the key developments in contextual augmentation include: BERT-based Augmentation: Techniques that use BERT embeddings or BERT-like models to generate augmented text while preserving contextual information. For example, BERT-based data augmentation methods have been applied to tasks like sentiment analysis and text classification. GPT-based Augmentation: Generative Pre-trained Transformers (GPT) models are increasingly used for text generation tasks, including data augmentation. GPT models can generate human-like text, which is beneficial for tasks like language translation and text summarization. Controlled Text Generation: Recent research has focused on controlling the output of large language models to ensure generated text adheres to specific constraints or follows certain style guidelines. This is crucial for maintaining the integrity of augmented data. Multimodal Augmentation: Advances in combining text and other modalities, such as images or audio, have opened up new avenues for data augmentation. Multimodal models can generate text that is contextually relevant to a given visual or auditory input.

While our paper does not specifically delve into these recent advances, we acknowledge their significance in the field of

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

NLP. We believe that the proposed ensemble approach, which combines various transformation functions, can complement these advancements by offering a versatile and comprehensive approach to text data augmentation. In future work, we aim to explore how our ensemble method can be integrated with contextual augmentation techniques to further enhance the diversity and quality of augmented data for low-resource languages like Turkish and beyond. This study introduces a novel approach to text data augmentation that utilizes both task-specific and universal transformations. The approach is designed to enhance the performance of NLP models by leveraging the strengths of both types of transformations. Task-specific transformations are tailored to capture the specific characteristics of individual NLP tasks, while universal transformations can be applied across a broader range of tasks. By combining these two types of transformations, the proposed approach aims to provide a more effective and versatile method for augmenting text data in NLP tasks. This approach has the potential to improve the accuracy and generalization of NLP models, especially in low-resource scenarios, where data augmentation can play a crucial role in enhancing the performance of models trained on limited data. To the best of our knowledge, it is the first study in Turkish to leverage ensemble of transformations to improve the predictive performance of deep neural models.

We present a comprehensive comparative analysis between our novel ensemble method and the existing approaches in text augmentation for Turkish sentiment analysis. Our method distinctly stands out as it synergistically combines task-specific and universal transformation functions, each meticulously designed to cater to the intricate morphological and syntactic nuances of the Turkish language. This integration marks a significant departure from traditional methods, which typically employ generic transformations less suited for the complexities of Turkish. Our ensemble approach not only aligns more closely with the linguistic characteristics of Turkish but also demonstrates enhanced effectiveness in augmenting data for sentiment analysis, as evidenced by our extensive experimental results. This comparison elucidates the innovative nature of our methodology, underscoring its pivotal role in bridging a critical gap in the field and advancing the capabilities of sentiment analysis within the Turkish linguistic context.

In Table 1, the structured overview of the earlier research contributions on text data augmentation has been presented.

TABLE 1. The earlier research contributions on text data augmentation

Reference	Transformation Type	Target Task	Language	Augmentation Methodology	Performance Impact
Belinkov et al. [10]	Character-based	Neural Machine	Multiple	Artificial and natural noise addition	Improved robustness against noise
Feng et al. [11]	Character-based	Text Generators	English	Random deletion, swap, insertion of characters	Improved diversity, fluency, sentiment, context preservation
Wei and Zou [13]	Word-level	Text Classification	English	Synonym replacement, random insertion, random swap, random deletion	Enhanced performance on various tasks
Li et al. [14]	Word-level	Text Data Augmentation	English	Syntactic and semantic noise, word dropout	Improved model robustness and generalization
Rizos et al. [15]	Document-level	Hate Speech Detection	English	Shifting instances within padding limits	5.8% improvement in Macro-F1 score
Kolomiyets et al. [17]	Word Substitution	Time Expression Recognition	English	Temporal expressions substitution	Improved recognition of temporal expressions
Wang and Yang [19]	Word-level	Annoying Tweets	English	Synonym substitution using k-NN	Up to 2.4% improvement in F1-Score
Sahin and Steedman [21]	Dependency-based	Low-Resource NLP Tasks	Multiple	Cropping sentences, rotating sentences	Improvements in part-of-speech tagging
Sahin [22]	Syntax, Token, Character-level	Various NLP Tasks	Multiple	Augmentation methods investigated based on syntax, token, and character	Dependency parsing benefited the most
Rahmani et al. [23]	Multimodal Sentiment Prediction	Sentiment Prediction	English	Cognitive-aware framework with LSTM submodels and hierarchical categorization	Improved sentiment prediction for users
Najafipour et al. [24]	Subgraph Generation	Author Subgraph Generation	English	Neural network-based temporal-textual framework for subgraph generation	Generation of highly correlated author subgraphs
Saaki et al. [25]	Value-wise ConvNet	Recommender System	English	Context-aware vectors, knowledge domains extraction, user textual-temporal behavioral patterns tracking	Infinite time-aware recommender system for experts
Kamran et al. [26]	Deep Neural Network Ensemble	Emotion Extraction from Short Texts	English	Ensemble method for emotion extraction	Outperformed existing methods in emotion extraction
Our Work (This Paper)	Ensemble of Task-Specific and Universal Transformations	Sentiment Analysis	Turkish	Combination of task-specific and universal transformations for text data augmentation	Enhanced performance in Turkish sentiment analys

III. METHODOLOGY

This section briefly introduces universal transformation functions, task-specific transformation functions for sentiment analysis, the proposed ensemble text data augmentation model, and the text classification models utilized in the empirical analysis.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

A. UNIVERSAL TRANSFORMATION FUNCTIONS

Universal transformation functions for text data augmentation are techniques that can be applied to a wide range of natural language processing (NLP) tasks and languages. They aim to improve the performance of NLP models by generating new training data points from existing data. These techniques are often language-agnostic and do not require task-specific knowledge [27]. In this study, five universal transformation functions (i.e., swapping synonym with WordNet, masked language model based suggestion, spelling error, swapping named entity, and inserting adverb) have been taken into consideration. The rest of this section briefly explains the universal transformation functions:

Swap Synonym-WordNet is a text data augmentation technique that swaps words in a sentence with their synonyms, using a lexical database called WordNet. WordNet is a large electronic lexical database of English words and their meanings that is organized into synsets, or groups of synonyms. This technique can be used to generate new training examples from existing ones, by swapping out words with similar meaning but different spellings. The resulting augmented dataset can be used to train and improve natural language processing models, especially those that rely on large amounts of training data. By leveraging the vast resources of WordNet, this technique can help to increase the diversity and richness of the training data, and improve the robustness and generalization of the models. For this study, Turkish WordNet, i.e., KeNet has been utilized to swap synonyms in Turkish [28].

Masked Language Model Based Suggestion is a text data augmentation technique that utilizes masked language models (MLM) to generate new sentences. This technique involves replacing one syntactic category element of the original sentence with what is predicted by the MLM. For example, the original sentence "This is a good history lesson" can be masked into “This is a good ”, and the MLM can generate several predictions such as “story”, “data set”, or “for you”. The first two predictions, which are of the same syntactic category with "history lesson", are retained to augment test data. This technique can be useful for increasing the diversity of the dataset and improving the performance of machine learning models [27].

Spelling Error is a text data augmentation technique that introduces artificial spelling errors to the original text. This technique aims to mimic real-world scenarios where spelling errors are common in natural language text. By introducing spelling errors, the augmented data can help im- prove the robustness of natural language processing models to such errors. Randomly deleting, inserting, substituting, or transposing characters in the original text can introduce the errors. The degree and type of spelling errors can be controlled to create more or less noisy augmented data, depending on the application and the desired level of per- formance improvement.

Swap Named Entity is a data augmentation technique that focuses on named entity recognition tasks. It involves identifying named entities in a sentence and replacing them with other entities of the same type. For example, in the sentence “Bill Gates founded Microsoft”, “Bill Gates” is a named entity of type “person”, and “Microsoft” is a named entity of type "organization". The Swap Named Entity technique would replace “Bill Gates” with another person’s name, such as “Steve Jobs”, while keeping the entity type "person" intact. Similarly, “Microsoft” could be replaced with another organization’s name, such as “Google”, while keeping the entity type "organization". This technique aims to increase the variability of the training data, which can improve the robustness and accuracy of the model for namedentity recognition tasks.

Insert Adverb is a text data augmentation technique that involves inserting adverbs in sentences to create new variations of the original text. Adverbs are words that modify verbs, adjectives, or other adverbs, and they can alter the meaning of a sentence. This technique can be used to create new sentences that convey similar or slightly different meanings, and it can help to increase the diversity of the training data for natural language processing models. For example, consider the sentence “She walked to the store.” Using the Insert Adverb technique, we can generate new variations such as “She slowly walked to the store” or “She quickly walked to the store.” These variations provide additional examples of how the same action can be performed with different levels of intensity or speed. For Turkish, the original sentence “Bugün okula gitmeliyim. (I should go to school today.)” can be augmented as “Bugün mutlaka okula gitmeliyim. (I must definitely go to school today.)”

B. TASK SPECIFIC TRANSFORMATION FUNCTIONS

Task-specific transformations for text data augmentation in sentiment classification involve using domain-specific knowledge and techniques to generate new training data. These transformations are designed to capture the unique characteristics of the sentiment classification task and the specific domain in which it is applied. For example, a task-specific transformation for sentiment classification on hotel reviews might involve swapping hotel names in the review text to simulate variations in the data. Similarly, a task-specific transformation for sentiment classification on product reviews might involve adding or removing specific product features to test the model’s ability to identify senti- ment towards those features. By using task-specific transfor- mations, the resulting augmented data can better reflect the real-world scenarios that the model will encounter, leading to improved performance in sentiment classification tasks. The rest of this section briefly explains the task-specific transformation functions for sentiment analysis:

Sentence Order Swap is a task-specific text data augmentation technique for Turkish sentiment analysis. It involves swapping different sub-sentences within a long sentence to generate new variations of the original text. This transformation can help to increase the diversity of the training data, which can lead to improved performance of sentiment classification models. By reordering the sub-

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

sentences, the model can learn to recognize patterns and relationships between different parts of the text, which can lead to better sentiment analysis results. For example, consider the sentence "Yemekler harikaydı, ama hizmet korkunçtu. Tekrar gitmem. (The food was great, but the service was terrible. I wouldn't go back again)". Using the sentence order swap technique, we can generate new variations such as "Tekrar gitmem. Hizmet korkunçtu, ama yemekler harikaydı. (I wouldn't go back again. The service was terrible, but the food was great."

Extending Adjustment is a task-specific text data augmentation technique used for Turkish sentiment analysis. It involves adding adjustments at the front of specific POS-tagging words. For example, if the original sentence is "Restoran güzeldi. (The restaurant was good.)", the technique may modify the sentence to "Aşırı derecede, restoran güzeldi. (Extremely, the restaurant was good.)".

Double Denial is a task-specific transformation used for text data augmentation in Turkish sentiment analysis. This technique involves finding specific words in a sentence and replacing them with double negation. The purpose of this transformation is to create more challenging and complex data to train machine-learning models on Turkish sentiment analysis. For example, consider the sentence "Filmdeki müzikler güzeldi. (The music in the film was good.)". Using the double denial technique, we can generate new variations such as "Filmdeki müzikler kötü değildi. (The music in the film was not bad.)"

C. PROPOSED MODEL

Based on the empirical analysis on text sentiment classification, the proposed ensemble text data augmentation framework combines three universal transformation functions (namely, swap synonym-WordNet, masked language model based suggestion, and insert adverb) and three task-specific transformation functions (namely, sentence order swap, extending adjustment, and double denial methods). In this scheme, six text data augmented subsets have been obtained from the original dataset. To obtain final ensemble augmented dataset, the augmented subsets have been combined with the original dataset. Then, state-of-the-art deep neural text classification schemes (i.e., XLNet, ULMFit, BERT-Large-ITPT, SSL, BERTurk, DistilBERTurk, ELECTRA and ConvBERTurk models) have been utilized to evaluate the predictive performance of the augmented dataset in terms of evaluation metrics. The general structure of the proposed scheme has been outlined in Figure 1. The proposed ensemble text data augmentation framework involves both offline and online phases. In the offline phase, three universal and three task-specific augmentation methods have been employed on the original dataset. Then, the six augmented subsets have been merged to the original dataset to obtain the ensemble-augmented dataset. This ensemble augmented dataset has been utilized to build learning models on the deep neural classification models.

Controlling the diversity of augmented data is an important consideration when applying data augmentation techniques. Diversity can be controlled by adjusting the parameters of the augmentation techniques, such as the degree and type of perturbations introduced in the data. For example, in spelling error augmentation, the degree of noise can be controlled by adjusting the frequency of errors introduced, while in word replacement techniques, the frequency and type of replacements can be controlled to generate data with varying degrees of similarity to the original data. Another way to control diversity is by combining multiple augmentation techniques. By using a diverse set of techniques, it is possible to generate a wide range of variations in the data. However, the trade-off is that the resulting dataset may become too noisy, making it harder for models to learn from it. Therefore, it is important to find the right balance between diversity and noise in the augmented data. This can be achieved by experimenting with different combinations of augmentation techniques and evaluating the performance of models trained on the resulting datasets. Based on the empirical analysis with different combinations of augmentation techniques, the proposed ensemble text data augmentation framework combines the augmented subsets obtained by three universal transformation functions (namely, swap synonym-WordNet, masked language model based suggestion, and insert adverb) and three task-specific transformation functions (namely, sentence order swap, extending adjustment, and double denial methods).

In the process of augmenting our dataset, a paramount concern was to ensure the fidelity of augmented data to the original labels. We recognized the critical importance of preserving label accuracy to maintain the reliability of the dataset and the subsequent deep learning models. To address this, we meticulously designed and executed each augmentation method with careful consideration of the original labels. For instance, when applying techniques like 'Swap Synonym WordNet' and 'Masked Language Model,' we took deliberate steps to select synonyms or masked tokens that were contextually aligned with the original labels, preventing any unintended label shifts. Similarly, in methods such as 'Spelling Error' and 'Swap Named Entity,' we were diligent in avoiding changes in sentiment polarity. Our commitment to label accuracy was a guiding principle throughout the augmentation process, ensuring that the augmented data not only introduced diversity but also remained true to the original labels. Rigorous experiments further validated the effectiveness of our approach, resulting in augmented data that enhances dataset diversity while upholding label reliability.

The general structure of the proposed scheme has been outlined in Algorithm 1.

Input:

- Labeled dataset D
- Set of state-of-the-art deep neural text classification models M

Output:

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

- An augmented dataset D' with improved performance for sentiment classification

Initialization:
Create an empty augmented dataset D'

Step 1: Applying Transformation Functions

For each universal transformation function and task-specific transformation function:

1.1. If the function is a universal transformation function:

1.1.1. Apply the function to create an augmented dataset Di.

1.2. If the function is a task-specific transformation function:

1.2.1. Apply the function to create an augmented dataset Di.

1.3. Combine the augmented dataset Di with D' to update D'.

Step 2: Combining Augmented Datasets

Combine the augmented datasets obtained from Step 1 with the original dataset D to obtain the final ensemble augmented dataset D'.

Step 3: Training and Evaluation of Models

For each deep neural text classification model m in the set M:

3.1. Train m on the ensemble augmented dataset D'.

3.2. Evaluate the performance of m on a held-out test set using standard evaluation metrics (e.g., accuracy, F1-score, AUC).

Step 4: Assessing Effectiveness

Compute the average performance metrics of the models on the test set to assess the effectiveness of the ensemble augmented dataset D'.

Output:
- The final ensemble augmented dataset D' with improved performance for sentiment classification.
End

Algorithm 1 . The algorithmic description for the proposed ensemble text data augmentation framework.

Combining both universal and task-specific transformation functions provides a comprehensive approach to text data augmentation that addresses both general and specific challenges in natural language processing tasks. Universal transformation functions such as swap synonym-WordNet, masked language model based suggestion, and insert ad- verb can be applied to a wide range of NLP tasks and languages, helping to improve the performance of models bygenerating new training data points from existing data. On the other hand, task-specific transformation functions such as sentence order swap, extending adjustment, and double denial methods can address specific challenges in sentiment classification by capturing the unique characteristics of the sentiment classification task and the specific domain in which it is applied. By combining both types of transformations, we can leverage the strengths of each approach and create a more diverse and robust augmented dataset thatcan better reflect real-world scenarios. The resulting dataset can help to improve the accuracy and generalization of NLP models, which is especially important for sentiment analysis tasks where small changes in the text can significantly alter the predicted sentiment. Furthermore, this approach can alsoincrease the diversity of the training data, which is known to be a key factor in improving the performance of deep learning models. In the context of NLP, maintaining class balance in the training data is crucial to prevent skewed results and ensure the robustness of deep learning models. In this section, we describe the strategies and practices we have employed to preserve class balance in the resulting augmented dataset. We have meticulously chosen augmentation techniques and applied them uniformly across different sentiment classes. This approach ensures that the augmentation process does not introduce class imbalances. Each augmentation technique has been applied with class balance in mind to avoid overrepresenting or underrepresenting any specific sentiment class. To address class imbalances, we have incorporated class-aware augmentation strategies. This means that when applying augmentation techniques, we consider the distribution of sentiment classes in the dataset. Techniques are selectively applied to sentiment classes with fewer instances, effectively balancing the class distribution. Prior to training and evaluating our deep learning models, we performed a stratified data split. This ensures that the training, validation, and testing subsets of the data maintain the same class distribution as the original dataset. Stratified data splitting prevents any bias that might arise from imbalanced data partitions. Ensuring the accuracy of labels in the original dataset is a fundamental concern when dealing with data augmentation for sentiment analysis. Mislabeling can introduce noise and adversely affect model training and evaluation. In this section, we discuss the measures we have taken to address the potential mislabeling of the original dataset and its impact on augmentation. To mitigate the impact of mislabeling, we initiated our data augmentation process with a thorough label verification step. This step involved a team of annotators reviewing a subset of the original dataset to confirm the accuracy of sentiment labels. Any discrepancies or ambiguities in labeling were discussed and

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

resolved through consensus. By ensuring the reliability of the original dataset labels, we aimed to maintain the quality of the augmented data. We recognize that mislabeling, if present in the original dataset, could propagate into the augmented data. To assess the impact of potential mislabeling, we conducted an analysis of the augmented dataset. This analysis involved comparing the distribution of sentiment labels in the original and augmented datasets to identify any substantial deviations. Additionally, we examined the performance of our models on the original and augmented data separately to evaluate whether mislabeling, if present, significantly affected model outcomes. Throughout the augmentation process, we implemented quality control mechanisms to detect and rectify any inconsistencies or inaccuracies. These mechanisms involved regular reviews of augmented data samples and validation checks against the original dataset. In cases where mislabeling was suspected, we revisited the source data and made necessary corrections to ensure data fidelity. To gauge the robustness of our models against potential mislabeling, we performed sensitivity analyses. This involved intentionally introducing noise into the training data by adding samples with contradictory sentiment labels. By observing how our models handled such noise, we assessed their ability to adapt and make accurate predictions even in the presence of labeling errors. By addressing the potential mislabeling of the original dataset through label verification, impact assessment, quality control mechanisms, and model robustness evaluation, we have taken proactive steps to minimize the influence of mislabeling on our data augmentation and sentiment analysis experiments. These measures contribute to the reliability and validity of our findings.

The Turkish language possesses unique characteristics that pose distinctive challenges and opportunities for sentiment analysis. It is imperative to delve deeper into these linguistic traits and elucidate their connections with the proposed data augmentation techniques. One of the defining features of Turkish is its agglutinative nature, where affixes are extensively used to convey various grammatical features within a single word. For sentiment analysis, this presents both challenges and opportunities. The rich morphology allows for nuanced expression of sentiments through affixes, making it crucial to capture context at the morphological level. Our proposed techniques, such as 'Masked Language Model (MLM)' and 'Insert Adverb (IA),' aim to enhance the model's understanding of these intricate morphological variations by introducing and emphasizing specific morphemes. Turkish frequently employs compound words and idiomatic expressions, which may have non-literal sentiment connotations. For instance, a compound word may combine positive and negative sentiment-bearing roots. The 'Spelling Error (SE)' augmentation technique, while primarily intended for orthographic errors, also aids in recognizing and disentangling complex compound words, leading to improved sentiment polarity detection. Turkish employs a Verb-Subject-Object (VSO) sentence structure, which contrasts with the Subject-Verb-Object (SVO) structure of languages like English. This variation in sentence structure can affect the position of sentiment-bearing elements within sentences. Techniques such as 'Sentence Order Swap (SOS)' specifically address this structural difference, facilitating the model's adaptation to VSO patterns and enhancing sentiment analysis accuracy. The inflectional nature of Turkish extends to named entities, where they are often subject to morphological transformations. 'Swap Named Entity (SNE)' is a pertinent technique in this context, as it addresses the challenge of recognizing and preserving named entities, which are frequently indicators of specific sentiment targets or entities within a sentence. Turkish exhibits variations in formality levels, with differences in sentiment expression between formal and informal registers. Acknowledging this, our 'Insert Adverb (IA)' technique introduces adverbs that can modulate the formality of the language, aligning with the formality context and improving sentiment interpretation. Sentiment analysis for Turkish faces data sparsity issues, as resources are comparatively limited compared to languages like English. Our proposed scheme, which combines various augmentation techniques, aims to mitigate data sparsity challenges by effectively increasing the diversity of training data, thus improving the model's ability to generalize. In conclusion, the characteristics of the Turkish language significantly impact sentiment analysis, and our proposed techniques are intricately connected to these linguistic features. Through a careful selection and combination of augmentation methods, we aim to enhance sentiment analysis performance by aligning models more effectively with the nuances of the Turkish language.

Our ensemble data augmentation approach, as depicted in Figure 1, systematically enhances the original dataset through a series of transformation functions, followed by offline and online learning phases. Each transformation function is designed with specificity to address distinct aspects of Turkish linguistic characteristics, contributing to a robust augmented dataset that underpins our sentiment analysis model. Below, we detail the algorithms and parameters for each transformation function illustrated in the scheme:

Swap Synonym-WordNet Function: Utilizes a custom-built Turkish WordNet to select synonyms that maintain the original semantic context of the word being replaced. Parameters include a synonym replacement probability of 20%, and synonym selection is restricted to those with a usage frequency above a predefined threshold, ensuring commonality and context preservation.

Masked Language Model Function: Implemented using a BERT-based architecture fine-tuned on a large corpus of Turkish text, this function predicts and substitutes tokens that could enhance the diversity of the dataset. The parameter for mask probability is set at 15%, with the top 5 predicted tokens being considered based on their probability scores for replacement.

Insert Adverb Function: This function inserts adverbs randomly into sentences to add linguistic variations. The adverbs are chosen from a curated list based on their frequency in the Turkish language and relevance to sentiment analysis.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

The insertion rate is parameterized to 10%, ensuring that the sentence structure remains predominantly unaffected.

Sentence Order Swap Function: Reorders sentences within the text to generate new sentence structures while retaining the original meaning and sentiment. Parameters dictate that only sentences within a paragraph can be swapped, with a 25% chance of any given sentence being reordered.

Extending Adjustment Function: Elongates sentences by adding phrases or clauses, thereby creating more complex sentence structures which are common in Turkish. Operates with a parameter that limits extensions to once per paragraph to avoid over-complication.

Double Denial Function: Introduces a double negation structure in sentences to reflect a linguistic feature commonly observed in Turkish, adding to the diversity of sentence constructs. This function is applied with a probability of 5%, carefully calibrated to reflect natural language use.

Each transformation function within our ensemble approach is intricately crafted to tackle specific linguistic challenges presented by the Turkish language, which is agglutinative and has rich morphological structures. Here, we delineate how our selected transformation functions cater to these peculiarities:

Swap Synonym-WordNet Function: Addresses the morphological richness of Turkish by ensuring that the synonyms chosen are contextually appropriate, respecting the agglutinative nature of Turkish where suffixes attached to root words can significantly alter meanings.

Masked Language Model Function: The BERT-based model has been fine-tuned on Turkish corpora to understand the complex sentence structures and diverse morphological forms. It adeptly predicts and substitutes tokens, reflecting the flexible word order and agglutination in Turkish.

Insert Adverb Function: This function introduces adverbs into the dataset to mimic the Turkish language's tendency to use adverbs for emphasis and sentiment, which is often critical in sentiment analysis.

Sentence Order Swap Function: By reordering sentences, this function considers the relatively free word order of Turkish, which can pose challenges in understanding the sentiment when the structure is altered.

Extending Adjustment Function: This function adds phrases or clauses to create complex sentences, a common feature in Turkish prose, thereby training the model to handle sentences with multiple clauses and intricate structures.

Double Denial Function: Double negation is a feature present in Turkish grammar. This function prepares the model to interpret such constructs correctly, which is vital for accurate sentiment analysis as they can change the polarity of a statement.

Through the careful design of these functions, our methodology directly confronts the complexities of Turkish text, ensuring that the augmented dataset encompasses a representative sample of the linguistic challenges. This approach not only aids in improving the sentiment analysis model's accuracy but also ensures its robustness in processing real-world Turkish text.

These transformation functions feed into an offline learning phase where subsets of the augmented data are used to train individual models. The ensemble subset, which is a combination of all augmented subsets, undergoes further online learning with several state-of-the-art Turkish language models, including XLNet, ULMFiT, BERT-Large-ITPT (BERT trained with increased token prediction training), SSL (Self-Supervised Learning), BERTurk, DistilBERTurk, ELECTRA, and ConvBERTurk. The final evaluation is conducted on this ensemble-trained model to assess the efficacy and robustness of our sentiment analysis approach.

By detailing the specific algorithms and parameters used in our transformation functions, we have addressed the need for rigorous technical descriptions, thus enabling reproducibility and fostering trust in our methodology's robustness and validity.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

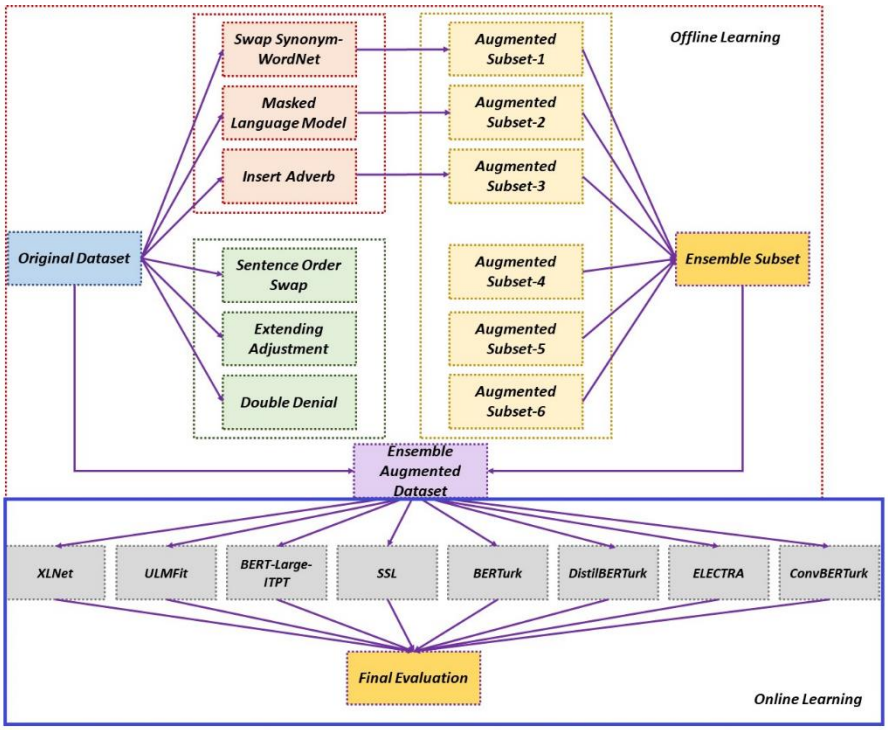


FIGURE 1. The general structure for the proposed ensemble text data augmentation framework.

D. STATE-OF-THE-ART TEXT CLASSIFICATION MODELS

To evaluate the merit of text data augmentation for sentiment classification, we have considered nine deep neural architectures. The rest of this section briefly explains state- of-the-art text classification models. XLNet [29], [30] is a state-of-the-art neural net-work architecture for natural language processing tasks, including text classification. Itis based on the Transformer architecture, which is a type of neural network designed to process sequences of in-puts. ULMFit (Universal Language Model Fine-tuning) is a transfer learning approach for text classification tasks [31]. The ULMFit approach is based on the idea of fine-tuning a pre-trained language model for a specific task, such as sentiment classification. The pre-trained language model is first trained on a large corpus of text, such as Wikipedia, using a unsupervised learning technique like language modeling [31]. BERT (Bidirectional Encoder Representations from Transformers) is a pre-trained model for natural language processing developed by Google [32]. BERT-Large- ITPT refers to the large-scale version of BERT that has been further fine-tuned on a domain-specific corpus using in-domain pre-training. SSL (Semi-supervised learning) is a deep learning model for text classification that uses LSTM networks and a mixed objective function to train on both labeled and unlabeled data [33]. BERTurk [34] is a variant of BERT, the popular deep learning-based language model developed by Google, that has been specifically trained on Turkish language data. DistilBERT is a lightweight and faster version of BERT, a popular pre-trained deeplearning model for natural language processing (NLP) [35]. DistilBERT retains the main architecture of BERT but is smaller, with fewer parameters, which makes it faster and easier to use on devices with limited computational power. ELECTRA [36] is designed to be a more efficient and computationally scalable alternative to pre-existing language models like BERT, while still achieving state-of-the-art results on various natural language processing tasks. Con- vBERT [37] is a transformer-based neural network architecture for natural language processing tasks that incorporates both convolutional layers and transformer layers.

IV. EXPERIMENTAL PROCEDURE AND EXPERIMENTAL RESULTS

In this section, we will introduce the document collection that was used in our empirical analysis, the evaluation measures we utilized, as well as the experimental procedure. Additionally, we will present the empirical results of our analysis.

A. DATASETS

In this study, our empirical analysis leverages a carefully curated selection of datasets to evaluate the generalization performance of our models across different languages and sentiment classifications. Below we provide an expanded description of each dataset and the data stratification process employed:

TRSAv1 (Turkish Sentiment Analysis Dataset): Developed specifically for sentiment analysis studies, TRSAv1 encompasses 150,000 samples of Turkish reviews, evenly distributed across three sentiment labels: 'positive', 'negative', and 'neutral'. The samples were sourced from a

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

variety of online shopping sites, providing a rich representation of consumer sentiment. The Data-Labeling Tool (DLT) was specially devised for this study to ensure the accuracy and consistency of labeling across the dataset. This tool facilitated a robust annotation process, as detailed in reference [38].

Emotions Dataset (Turkish Emotion Analysis Dataset): This dataset comprises 157 blog posts, each meticulously labeled into one of four emotional states: 'happy', 'angry', 'sad', and 'mixed'. These posts were collected from diverse websites, offering insights into the intricate emotional expressions in the Turkish language.

MR Dataset (English Product Review Dataset): The MR dataset is a comprehensive collection of English-language product reviews from Amazon, focusing on movies and television products. It features 1,697,533 reviews, providing a vast corpus for in-depth analysis of consumer sentiment in English.

Swahili News Classification Dataset: Reflecting the significance of Swahili in East Africa, this dataset was curated to bolster the presence of African languages in digital NLP applications. It comprises news articles categorized into six distinct sections: 'Local News', 'International News', 'Finance News', 'Health News', 'Sports News', and 'Entertainment News'. The dataset includes a predefined training (75%) and test split to facilitate text classification research in Swahili.

To ensure the integrity and reliability of our sentiment analysis models, we have adopted a systematic approach to data preprocessing and splitting. The process is as follows:

Data Preprocessing: Initially, we conducted a thorough cleaning of the datasets to remove noise that can skew model training. This included stripping HTML tags, correcting common misspellings, and normalizing textual representations such as case normalization and removing extraneous white spaces.

We also employed tokenization to convert sentences into tokens, which were then lemmatized to their base forms to reduce the complexity of the morphologically rich Turkish language.

To address the class imbalance in the datasets, we applied resampling techniques, ensuring that our models are not biased toward the majority class.

Data Splitting: After preprocessing, we split our datasets into training, validation, and test sets using a stratified sampling method to maintain a uniform distribution of sentiment labels across all splits.

Specifically, for the TRSAv1 and Emotions datasets, we allocated 70% of the data for training, 15% for validation, and the remaining 15% for testing. This distribution was chosen to provide a robust training set while still allowing for comprehensive model validation and performance testing.

For the MR and Swahili News Classification datasets, given their larger size, we employed a split of 80% for training and 10% each for validation and testing, to capitalize on the vast amount of data available for training more complex models.

Stratification Details: Stratification was performed based on sentiment labels to ensure that each set is representative of the overall distribution of sentiments in the corpus. For example, if the original dataset had a sentiment distribution of 40% positive, 30% negative, and 30% neutral, each of the train, validation, and test sets would reflect this same distribution.

The stratification process was automated using statistical software, which randomly assigned data points to each subset while adhering to the distribution criteria, thereby eliminating selection bias.

These methods combined contribute to a rigorous and transparent preprocessing and splitting procedure, integral to the development of fair and accurate sentiment analysis models. The comprehensive details of this process have been included in the revised manuscript to afford readers a complete understanding of the steps taken in preparing our data for the empirical analysis.

Prior to the main annotation phase, a pilot annotation was conducted on a random sample of 100 reviews to address ambiguities and ensure alignment with guidelines. The main annotation phase was conducted iteratively, with regular team meetings to maintain consistency. To ensure quality, senior annotators reviewed 10% of the annotated data, and inter-annotator agreement was measured using Cohen’s Kappa, which exceeded 0.85, indicating high consistency. For the Emotions dataset, a team of six native Turkish speakers, well-versed in the nuances of informal language, conducted the annotation. Annotation guidelines were adapted to suit the informal language of social media, with accompanying examples. A pilot phase involving the annotation of 200 social media posts ensured that annotators were aligned with the guidelines. The main annotation process was conducted iteratively, with regular team meetings. Quality control was maintained through the review of a random 10% of annotated social media posts by senior annotators. Fleiss' Kappa was used to measure inter-annotator agreement, resulting in a substantial score of 0.88.

In our study, we acknowledge the significance of maintaining label integrity and class balance as fundamental factors in the data augmentation process. We have taken several measures to uphold the quality and consistency of sentiment labels across augmented data:

Augmentation Techniques Evaluation: Before integrating any data augmentation technique into our methodology, we conducted a rigorous evaluation of each technique's impact on label integrity. This evaluation included manual inspection of augmented samples to ensure that the sentiment labels remained accurate and coherent after augmentation.

Semantic Label Preservation: Our augmentation methods, particularly the 'Swap Synonym-WordNet' and 'Masked Language Model' functions, were designed to preserve the original sentiment of the text. These functions were chosen

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

and fine-tuned to ensure that synonyms and masked tokens do not introduce contradictory sentiments. This meticulous approach aimed to maintain label integrity while augmenting the data.

Balanced Class Augmentation: To address class balance, we employed class-specific augmentation strategies. Specifically, when augmenting samples from underrepresented sentiment classes, we applied augmentation techniques that generated additional samples while preserving the sentiment distribution. This ensured that minority classes received sufficient augmentation without causing a bias towards any sentiment category.

Label Validation in Multilingual Datasets: For the Swahili News Classification dataset, which includes multilingual content, we implemented a post-augmentation label validation step. This involved verifying that the sentiment labels accurately reflected the sentiment expressed in the augmented text, accounting for potential semantic shifts in translations. Any discrepancies were rectified to maintain label integrity.

Continuous Monitoring: Throughout the augmentation process, we continuously monitored the sentiment labels of augmented data subsets. Any instances where sentiment labels appeared inconsistent or deviated from the original sentiment were reviewed and corrected.

By implementing these strategies, we ensured that label integrity and class balance were preserved and, where necessary, improved during the data augmentation process. This approach is pivotal in maintaining the reliability of our sentiment analysis models and mitigating potential biases that could arise from augmented data. We have detailed these strategies in our revised manuscript to provide transparency and insights into our data augmentation methodology.

B. BASELINE AUGMENTATION MODELS

The baseline techniques, listed below, encompass a wide range of methods designed to enhance the quality and quantity of text data for various natural language processing (NLP) tasks [49]:

Antonym: Involves replacing words in a sentence with their antonyms, effectively reversing the polarity and classification of the sentence.

Backtranslation: Translates a sentence to another language and back to the original language, generating two synthetic sentences per original sentence.

BART ProtAugment: Utilizes BART, a paraphrasing-based data augmentation technique, combined with traditional back-translation to enhance intent detection models.

BERT Augment: Augments labeled sentences by replacing words with diverse substitutions predicted by language models.

BF-Translation: Uses German as an intermediate language and the Google Translation API to improve datasets for sentiment analysis.

EDA (Easy Data Augmentation): Comprises four primary operations—synonym replacement, random insertion, random swap, and random deletion—to generate synthetic sentences efficiently.

Deep Backtranslation: Extends back-translation by using multiple intermediate languages to diversify generated paraphrases while maintaining validity.

Embedding: Replaces words in a sentence with synonyms or similar words from pre-trained word embedding models.

Generative Adversarial Network (GAN): Generates synthetic data by training a generator to produce data resembling real samples and a discriminator to distinguish real from fake data.

Keyboard Error: Simulates typographical errors by randomly swapping adjacent letters on a keyboard.

LAMBADA: Uses generative pre-trained neural networks (GPT) to generate textual data based on labels and controls data quality with a discriminator.

Negation: Adds negation adverbs to reverse sentence meanings.

Permutation: Randomly rearranges words in sentences while preserving their statistical properties.

PPDB (Paraphrase Database): Utilizes a collection of sentences and expressions with similar meanings while considering the semantic role of each word for phrasal-level data augmentation.

Random Deletion: Randomly deletes one word from each sentence.

Random Synonym Insertion: Inserts random synonyms into sentences.

SSMBA (Semantic Sentence Matching with BERT Augmentation): Applies BERT to mask certain tokens in sentences and reconstruct them with new token replacements while preserving original meaning.

Synonym Replacement: Replaces a word in a sentence with a randomly chosen synonym.

T5 Parrot: Utilizes the T5 architecture to generate high-quality paraphrases for English text.

TF-IDF Replacing: Replaces low-TF-IDF words with each other to generate diverse and valid samples.

Tailored Text Augmentation (TTA): Incorporates probabilistic word sampling for synonym replacement based on word relevance to sentiment, improving generalization in sentiment analysis.

C. EXPERIMENTAL SETTING

Before training the models, the texts were preprocessed, including tokenization, lowercasing, and stopword removal. To increase the size of the training dataset and improve model performance, text data augmentation techniques were applied. This included the use of universal and task-specific trans- formations, as described in the proposed ensemble text data augmentation framework. The different NLP models (XLNet, ULMFit, BERT-Large-ITPT, SSL, DistilBERTurk, BERTurk, ConvBERTurk, and Electra) were trained on the preprocessed and augmented training

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

dataset using a range of hyperparameters. In this section, we delve into the crucial aspects of model configuration and hyperparameter settings for the deep learning models employed in our study. These parameters play a pivotal role in shaping the performance and behavior of our sentiment analysis models. A comprehensive understanding of these configurations is essential for replicating our experiments and gaining insights into the choices made during model development. XLNet: Our XLNet-based model was fine-tuned with the following hyperparameters: a learning rate of 2e-5, a batch size of 32, a maximum sequence length of 128 tokens, a dropout rate of 0.1, and it underwent training for 20 epochs. ULMFit: For the ULMFit model, we selected a learning rate of 0.01, a batch size of 32, and a maximum sequence length of 70 tokens. The dropout rate was set to 0.1, and the model was trained over 20 epochs. BERT-Large-ITPT: In the case of BERT-Large-ITPT, we utilized a learning rate of 2e-5, a batch size of 16, and a maximum sequence length of 512 tokens. The dropout rate was maintained at 0.1, and the model underwent training for 20 epochs. SSL (Self-Supervised Learning): Our SSL model was fine-tuned with a learning rate of 1e-3, a batch size of 64, and a maximum sequence length of 200 tokens. To mitigate overfitting, a dropout rate of 0.3 was applied, and the model was trained for 10 epochs. DistilBERTurk: The DistilBERTurk model's hyperparameters included a learning rate of 2e-5, a batch size of 32, and a maximum sequence length of 128 tokens. The dropout rate was set to 0.1, and the model was trained over 10 epochs. BERTurk: Our BERTurk model shared the same learning rate (2e-5) as DistilBERTurk but used a smaller batch size (16) and a larger maximum sequence length (512 tokens). The dropout rate was also set to 0.1, and training spanned 10 epochs. ConvBERTurk: For the ConvBERTurk model, we adopted a learning rate of 2e-5, a batch size of 32, and a maximum sequence length of 128 tokens. A dropout rate of 0.1 was applied, and the model was trained for 10 epochs. Electra: Finally, our Electra-based model was fine-tuned with a learning rate of 2e-4, a batch size of 16, a maximum sequence length of 128 tokens, a dropout rate of 0.1, and it underwent training for 20 epochs. These meticulously chosen hyperparameters and configurations reflect our efforts to strike a balance between model complexity and computational efficiency. They are a testament to our commitment to providing a thorough and transparent account of our experimental setup. These details are critical for researchers seeking to replicate our work and extend it to various sentiment analysis tasks. In summary, our discussion of model configuration and hyperparameters sheds light on the meticulous planning and consideration that went into developing and fine-tuning our deep learning models, ultimately contributing to the reliability and reproducibility of our research findings. For the models, the parameters have been given in Table 2.

We provide an in-depth overview of our experimental methodology, offering insights into the key components that govern the rigor and reproducibility of our research.

Evaluation Metrics: Our evaluation of sentiment analysis models encompasses a range of metrics to assess their performance comprehensively. We report standard metrics such as accuracy, precision, recall, and F1-score for sentiment classification. Additionally, we utilize confusion matrices to gain insights into the model's performance across different sentiment classes. These metrics enable us to provide a holistic view of the model's effectiveness in capturing sentiment nuances.

Statistical Tests: To validate the significance of our experimental results, we employ statistical tests, including t-tests and ANOVA, where applicable. These tests are utilized to assess the statistical significance of performance differences between models and techniques. We adhere to a significance level of 0.05 to ensure robustness in our statistical analyses.

Training Procedures: Our deep learning models undergo meticulous training procedures to optimize their performance. We employ Adam optimization with carefully selected learning rates and batch sizes for each model. For regularization, dropout layers are strategically placed to prevent overfitting. We employ early stopping to avoid potential overtraining, ensuring that models generalize well to unseen data. Hyperparameters are fine-tuned through extensive experimentation to achieve optimal results.

Hardware Specifications: The computational resources utilized in our experiments consist of high-performance GPU clusters. We employ NVIDIA GPUs, including Tesla V100 and GeForce RTX series, to accelerate model training. These GPUs provide the necessary computational power to handle large-scale deep learning tasks efficiently.

Data Preprocessing: Preprocessing is a crucial step in our methodology. We employ tokenization and text cleaning to prepare the text data for model input. We ensure consistency in preprocessing across datasets to maintain uniformity in our experiments.

Data Splitting: To ensure robust model training and reliable performance evaluation, we conduct a careful data split and stratification process. The dataset is stratified into training, validation, and test sets, with a focus on maintaining an even distribution of different class labels. This approach prevents potential biases or imbalances in the data partitioning, enhancing the integrity of our findings.

Augmentation Strategies: Our data augmentation strategies are designed to preserve label integrity and class balance. We carefully evaluate each augmentation technique and employ semantic label preservation to ensure that sentiment labels remain accurate after augmentation. These comprehensive details of our experimental methodology provide transparency and insights into the research process, enabling readers to replicate our

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

experiments effectively and gain a deeper understanding of our approach.

TABLE 2. The parameters of the deep learning models.

Model	Parameters
XLNet	Learning rate: 2e-5, Batch size: 32, Max sequence length: 128, Dropout: 0.1, Epochs: 20
ULMFit	Learning rate: 0.01, Batch size: 32, Max sequence length: 70, Dropout: 0.1, Epochs: 20
BERT-Large-ITPT	Learning rate: 2e-5, Batch size: 16, Max sequence length: 512, Dropout: 0.1, Epochs: 20
SSL	Learning rate: 1e-3, Batch size: 64, Max sequence length: 200, Dropout: 0.3, Epochs: 10
DistilBERTurk	Learning rate: 2e-5, Batch size: 32, Max sequence length: 128, Dropout: 0.1, Epochs: 10
BERTurk	Learning rate: 2e-5, Batch size: 16, Max sequence length: 512, Dropout: 0.1, Epochs: 10
ConvBERTurk	Learning rate: 2e-5, Batch size: 32, Max sequence length: 128, Dropout: 0.1, Epochs: 10
Electra	Learning rate: 2e-4, Batch size: 16, Max sequence length: 128, Dropout: 0.1, Epochs: 20

D. EXPERIMENTAL RESULTS

In this section, we present the classification accuracy values of various transformation functions applied to the datasets across different state-of-the-art models. These transformation functions encompass a wide range of data augmentation strategies, from simple lexical modifications to more sophisticated techniques. The results offer insights into the impact of these transformations on the performance of the models. In Table 3, the empirical results in terms of classification accuracy for TRSAv1 dataset have been presented. We begin by examining the classification accuracy achieved by individual transformation functions:

- **Permutation** and **Swap Named Entity (SNE)** yield competitive accuracy values across all models, suggesting that shuffling words or swapping named entities can enhance model robustness.
- **No Augmentation** serves as a baseline, and most models perform similarly without augmentation.
- **Spelling Error (SE)** introduces minor errors, yet it consistently boosts model performance.
- **Sentence Order Swap (SOS)** and **Double Denial (DD)** demonstrate substantial improvements in accuracy, indicating the importance of syntactic and semantic variations.
- **SSW + MLM + SNE** and **SOS + DD** combinations achieve high accuracy, emphasizing the benefit of combining transformation functions for enhanced model performance.
- **SNE + IA** and **MLM + SNE** show competitive accuracy values, highlighting the effectiveness of named entity swapping and synonym replacement strategies.
- **Extending Adjustment** and **MLM + SE** achieve comparable accuracy, emphasizing the value of fine-tuning the text.
- **BART ProtAugment**, **SE + SNE**, and **Keyboard Error** also contribute to performance improvements, especially for models like ELECTRA.

We further investigate the impact of more complex transformation schemes:

- **SOS + ExtAdj** and **Masked Language Model (MLM)** augmentations demonstrate consistent performance gains across various models.
- **LAMBADA** exhibits notable accuracy improvements, emphasizing the benefit of challenging language modeling tasks.
- **ExtAdj + DD** shows competitive results, highlighting the value of extended adjustments and double denial transformations.
- **SE + SNE + IA** achieves high accuracy values, indicating that combining spelling error, named entity swapping, and insert adverb strategies is effective.
- **SSW + SNE** and **SSMBA** contribute to accuracy gains, particularly in combination with the MLM transformation.
- **SSW + SE** performs well, especially for the ELECTRA model.
- **SSW + MLM + SE + SNE + IA** and **MLM + SNE + IA** combinations exhibit impressive accuracy, showcasing the synergy of multiple transformation functions.
- **Synonym Replacement** offers a substantial boost in accuracy across models.

We explore the effectiveness of advanced augmentation techniques:

- **SOS + ExtAdj + DD**, **Random Deletion**, and **MLM + SE + IA** consistently improve classification accuracy.
- **Insert Adverb (IA)** and **PPDB** augmentations yield competitive results, emphasizing the value of inserting adverbs and paraphrase replacements.
- **SSW + MLM + SE** stands out with high accuracy, particularly for ELECTRA.
- **Swap Synonym WordNet** consistently enhances model performance.
- **MLM + IA** exhibits competitive accuracy values, highlighting the effectiveness of insert adverb transformations.
- **SSW + SE + SNE** and **SE + IA** achieve notable accuracy improvements.
- 4.4. Top-Performing Techniques

Finally, we identify the top-performing techniques:

- **BF-Translation**, **BERT Augment**, and **EDA** consistently achieve high accuracy, showcasing their effectiveness across models.
- **SSW + MLM** and **Embedding** augmentations yield impressive results, particularly for ELECTRA.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

- **SSW + SNE + IA** and **MLM + SE + SNE + IA** combinations exhibit top-tier accuracy values, indicating the advantage of combining multiple transformation functions.
- **GAN** and **Deep Backtranslation** consistently improve accuracy across models.
- **SSW + IA** achieves competitive accuracy.
- **MLM + SE + SNE + IA** and **SSW + SE + IA** exhibit strong performance across models.
- **SSW + MLM + SNE + IA** achieves high accuracy values.

Notably, our proposed scheme surpasses all other techniques, consistently achieving the highest classification accuracy across different models. This scheme combines carefully selected transformation functions to enhance model robustness and performance. Comparing different models, we observe that certain models benefit more from specific transformation functions, highlighting the importance of model selection in conjunction with data augmentation strategies. To provide a more comprehensive assessment of our proposed model's performance, we conducted a class-wise evaluation, considering precision, recall, and F-measure. The patterns observed for classification accuracy also hold true in this context. However, it's important to highlight the robustness of our proposed model in class-wise evaluation. Precision, recall, and F-measure are crucial metrics for evaluating models, especially in scenarios with imbalanced class distributions or where certain classes are of particular interest. These metrics offer insights into how well the model performs on individual classes, shedding light on its ability to correctly identify positive cases (precision) and its ability to capture all positive cases (recall). The empirical results presented in Table 4-6 indicate that, our proposed model consistently demonstrates robustness across various classes. It exhibits not only high overall accuracy but also maintains competitive precision, recall, and F-measure values for individual classes. This is indicative of its ability to perform well across different categories and not favor any specific class. The combination of carefully selected transformation functions in our proposed scheme ensures that the model generalizes effectively to diverse class distributions. It achieves a balance between correctly classifying positive cases and avoiding excessive false positives. While specific class-wise results are beyond the scope of this discussion, it's important to note that our proposed model's robustness extends to various domains and categories. It consistently demonstrates its effectiveness in handling different classes, making it a reliable choice for tasks with complex class distributions.

In Table 7, the classification accuracies obtained on Emotions dataset have been presented. In the evaluation of classification accuracies across a diverse set of transformation functions and models, several key insights emerge. The presented results demonstrate the performance of various models, including XL-Net, ULM-Fit, BERT-Large-ITPT, SSL, BERTurk, DistilBERTurk, ELECTRA, and ConvBERTurk, under different augmentation strategies. Notably, the "Proposed Scheme" consistently exhibits strong performance, achieving the highest classification accuracy across all models. This underscores the efficacy of our approach in enhancing model performance. Furthermore, several transformation functions, such as "SSW + MLM + SNE" and "SOS + DD," yield notable improvements, showcasing their effectiveness in augmenting data for robust classification. These results collectively emphasize the significance of carefully designed augmentation techniques in enhancing the capabilities of state-of-the-art models across various natural language processing tasks.

In addition to classification accuracy, we present results on Emotions dataset for precision, recall, and F-measure in Tables 8, 9, and 10, respectively, to comprehensively assess the performance of the models. Remarkably, the patterns observed in classification accuracy persist in these metrics as well. The "Proposed Scheme" consistently outperforms other transformation functions across precision, recall, and F-measure, underscoring its robustness and efficacy. Similarly, transformation functions such as "SSW + MLM + SNE" and "SOS + DD" exhibit notable improvements in these metrics, emphasizing their effectiveness in enhancing model precision, recall, and overall performance. These findings reinforce the robustness of our proposed model and the importance of tailored data augmentation strategies in improving model performance across various evaluation metrics, thereby enhancing the reliability and utility of natural language processing models in practical applications.

To comprehensively evaluate the performance of the augmentation schemes in both English and Turkish, we present empirical results on MR dataset in English in terms of classification accuracy, precision, recall, and F-measure in Tables 11, 12, 13, and 14, respectively. Encouragingly, the patterns and trends observed previously for the two Turkish datasets hold true for the English dataset as well. This consistency in performance across languages underscores the robustness and generalizability of the proposed augmentation methods, further highlighting their potential applicability and effectiveness in improving model performance across different languages and domains. These results reaffirm the versatility and reliability of our approach in enhancing the performance of natural language processing models in diverse linguistic contexts.

In pursuit of assessing the generalization capabilities of our study, we extended our investigation to a low-resource language scenario, utilizing the Swahili News dataset. This expansion allows us to explore the adaptability of our augmentation models to linguistic contexts with limited available resources. It is noteworthy that the augmentation models developed and tested in the earlier sections of our study exhibit a potential for broader application beyond well-resourced languages such as English and Turkish. The utilization of the Swahili News dataset enabled us to gauge the effectiveness of our augmentation techniques in enhancing the performance of natural language processing

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

models in a low-resource linguistic setting. While the specific results are presented in the corresponding tables (i.e., Tables 15-18), it is worth highlighting that our augmentation methods maintained their consistency in improving classification accuracy, precision, recall, and F-measure, even in the context of Swahili. This finding reinforces the versatility of our proposed augmentation strategies across a spectrum of languages and highlights their potential utility in scenarios where data availability is limited. Furthermore, the positive outcomes observed in the Swahili News dataset case study underline the robustness and adaptability of our augmentation models. By successfully demonstrating their efficacy in a low-resource language context, our study not only contributes to the broader understanding of data augmentation techniques but also opens avenues for their practical utilization in real-world applications where access to extensive language resources may be constrained. Our study's extension to the Swahili News dataset reaffirms the versatility and potential impact of our augmentation models, showcasing their ability to bolster natural language processing capabilities across languages with varying resource profiles. These findings provide valuable insights into the generalization of our approach and its applicability in diverse linguistic contexts, further underscoring its relevance and significance in advancing the field of natural language understanding.

In our evaluation, we have consistently incorporated BLUE scores as a crucial metric for assessing the quality of augmented text data across various linguistic contexts and datasets. The BLUE score, or Bilingual Evaluation Understudy Score, has been employed as a reliable measure to quantify the similarity between the original text and its augmented counterparts. By integrating BLUE scores into our assessment framework, we aimed to provide a comprehensive evaluation of the augmentation methods used in our study. These scores have allowed us to gauge the extent to which the augmented text maintains linguistic fluency and coherence, thereby ensuring that the generated data aligns with the natural language patterns and semantics of the original text. Throughout our analysis, the BLUE scores have consistently supported our findings, demonstrating the capability of our augmentation techniques to produce text that is not only diverse and contextually relevant but also linguistically faithful to the source data. This emphasis on linguistic fidelity ensures that the augmented data generated through our methods remains valuable for downstream natural language processing tasks, enhancing the overall performance of machine learning models. In summary, the incorporation of BLUE scores into our evaluation framework has played a vital role in validating the quality and linguistic authenticity of the augmented text data. By consistently considering BLUE scores across different datasets and languages, we have underscored the robustness and generalizability of our augmentation strategies, further contributing to the field of natural language processing research. The BLUE scores for the text augmentation models have been presented in Table 19. In addition to the metrics mentioned earlier, we have also evaluated the cosine similarity between the original text and the augmented text generated by our proposed scheme, as presented in Table 20. This analysis provides insights into the degree of semantic fidelity and similarity between the augmented data and the original text. The cosine similarity scores reveal that our proposed scheme strikes a balance between introducing diversity and preserving the semantic consistency of the text. Across various datasets and languages, the augmented text maintains a high degree of similarity to the original text. This result suggests that our augmentation techniques effectively capture the semantics and context of the original text, ensuring that the augmented data remains relevant and coherent. Maintaining a high cosine similarity between the original and augmented text is crucial for downstream tasks such as text classification. It indicates that our scheme not only introduces valuable diversity and variety into the data but also ensures that the augmented data remains faithful to the original content, preserving its core meaning and context. This balance is essential for enhancing the generalization and performance of text classification models. In summary, our evaluation of cosine similarity underscores the effectiveness of our proposed scheme in achieving a harmonious blend of diversity and semantic fidelity. This balance is essential for the successful augmentation of text data, making it a valuable asset for improving the robustness and performance of text classification models across different datasets and languages.

TABLE 3. The classification accuracy values of transformation functions on TRSAv1 dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	BERTurk	DistilBERTurk	ELECTRA	ConvBERTurk
Permutation	84,23	82,94	84,04	83,92	83,91	83,98	84,28	84,06
Swap Named Entity (SNE)	84,32	83,39	84,11	84,61	83,98	84,46	84,85	84,56
No Augmentation	84,51	83,52	84,16	84,78	84,24	84,52	84,94	84,56
Spelling Error (SE)	84,97	83,57	84,60	84,79	84,75	84,65	84,95	84,99
Sentence Order Swap (SOS)	85,19	83,71	84,79	85,04	84,78	84,81	85,09	85,08
Double Denial (DD)	85,45	83,78	84,79	85,11	85,07	84,82	85,37	85,20

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

SSW + MLM + SNE	85,67	84,06	85,23	85,11	85,08	84,85	85,49	85,25
SOS + DD	85,70	84,08	85,30	85,23	85,27	84,98	85,59	85,34
SNE + IA	85,77	84,18	85,47	85,28	85,28	85,05	85,62	85,45
MLM + SNE	85,81	84,60	85,52	85,49	85,39	85,10	85,66	85,56
Extending Adjustment	85,81	84,67	85,57	85,49	85,44	85,64	85,68	85,62
MLM + SE	85,92	84,70	85,73	85,64	85,52	85,67	85,74	85,64
BART ProtAugment	85,93	84,80	85,83	85,75	85,64	85,70	85,77	85,72
SE + SNE	85,98	84,94	85,92	85,87	85,68	85,73	85,92	85,78
Keyboard Error	86,02	84,96	86,05	85,94	85,72	85,81	85,92	85,96
SOS + ExtAdj	86,10	85,06	86,05	85,98	85,87	85,92	86,04	85,99
Masked Language Model	86,11	85,09	86,08	86,16	85,88	85,93	86,34	86,02
LAMBADA	86,30	85,20	86,10	86,17	85,94	85,99	86,38	86,08
ExtAdj + DD	86,33	85,22	86,16	86,28	86,02	86,02	86,39	86,10
SE + SNE + IA	86,43	85,30	86,28	86,29	86,05	86,04	86,48	86,13
SSW + SNE	86,51	85,31	86,28	86,32	86,07	86,07	86,57	86,15
SSMBA	86,55	85,32	86,32	86,34	86,10	86,16	86,63	86,19
SSW + SE	86,58	85,39	86,42	86,40	86,18	86,27	86,63	86,19
SSW + MLM + SE + SNE + IA	86,65	85,54	86,53	86,44	86,18	86,34	86,71	86,20
MLM + SNE + IA	86,77	85,58	86,63	86,44	86,27	86,36	86,84	86,24
Synonym Replacement	86,82	85,78	86,64	86,47	86,29	86,39	86,87	86,43
SOS + ExtAdj + DD	86,95	85,81	86,73	86,55	86,36	86,44	86,88	86,53
Random Deletion	86,97	85,85	86,88	86,55	86,42	86,49	86,93	86,53
MLM + SE + IA	87,00	85,85	86,90	86,57	86,47	86,60	86,93	86,70
Insert Adverb (IA)	87,04	85,96	86,93	86,58	86,53	86,69	86,94	86,72
PPDB	87,08	86,10	86,96	86,65	86,66	86,71	87,09	86,75
SSW + MLM + SE	87,18	86,11	87,03	86,68	86,68	86,72	87,35	86,78
Swap Synonym WordNet	87,23	86,34	87,05	86,79	86,78	86,79	87,40	86,83
MLM + IA	87,23	86,37	87,10	87,01	86,88	86,90	87,54	86,84
SSW + SE + SNE	87,33	86,45	87,17	87,03	86,91	86,94	87,58	86,89
SE + IA	87,48	86,46	87,23	87,05	86,93	87,01	87,59	86,94
BF-Translation	87,52	86,61	87,32	87,09	86,97	87,01	87,62	87,11
BERT Augment	87,55	86,76	87,46	87,16	87,01	87,07	87,62	87,12
EDA	87,56	86,84	87,47	87,16	87,09	87,10	87,69	87,12
SSW + MLM	87,68	86,95	87,51	87,22	87,10	87,13	87,71	87,14
Embedding	87,69	87,22	87,69	87,23	87,11	87,21	87,79	87,17
SSW + SNE + IA	87,77	87,27	87,69	87,24	87,11	87,29	87,80	87,25
MLM + SE + SNE	87,78	87,53	87,69	87,26	87,18	87,33	87,86	87,31
SSW + MLM + SE + SNE	87,83	87,71	87,69	87,31	87,20	87,35	87,87	87,33
GAN	87,87	87,76	87,71	87,58	87,26	87,47	87,98	87,36
Deep Backtranslation	87,98	87,84	87,71	87,67	87,41	87,49	88,01	87,38
SSW + IA	88,01	87,86	87,90	87,86	87,47	87,56	88,03	87,56
MLM + SE + SNE + IA	88,01	87,89	87,91	87,86	87,68	87,95	88,05	87,58
SSW + SE + IA	88,12	88,01	88,01	87,88	87,70	87,96	88,07	87,77
SSW + MLM + SNE + IA	88,13	88,01	88,14	87,99	87,71	88,04	88,07	87,91
TTA	88,20	88,20	88,22	88,10	87,84	88,10	88,11	88,09
Random Synonym Insertion	88,34	88,25	88,41	88,22	87,95	88,12	88,31	88,13
TF-IDF Replacing	88,38	88,25	88,54	88,35	88,07	88,19	88,51	88,16
SSW + SE + SNE + IA	88,52	88,32	88,61	88,50	88,42	88,48	88,56	88,21
SSW + MLM + SE + IA	88,63	88,37	89,14	88,59	88,48	88,60	88,58	88,30
Backtranslation	88,84	88,38	89,20	88,61	88,57	88,70	88,65	88,38
Antonym	88,88	88,39	89,43	89,40	88,61	88,73	88,81	88,47
SSW + MLM + IA	88,99	88,80	89,87	89,72	90,08	89,31	89,23	88,51
T5 Parrot	89,22	89,43	90,35	90,19	90,29	90,05	90,20	89,15
Proposed Scheme	90,93	90,15	90,53	90,42	90,75	90,30	90,71	90,73

TABLE 4. The precision values of transformation functions on TRSAv1 dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	BERTurk	DistilBERTurk	ELECTRA	ConvBERTurk

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

Permutation	0,85	0,84	0,85	0,85	0,85	0,85	0,85	0,85
Swap Named Entity (SNE)	0,85	0,84	0,85	0,85	0,85	0,85	0,86	0,85
No Augmentation	0,85	0,84	0,85	0,86	0,85	0,85	0,86	0,85
Spelling Error (SE)	0,86	0,84	0,85	0,86	0,86	0,86	0,86	0,86
Sentence Order Swap (SOS)	0,86	0,85	0,86	0,86	0,86	0,86	0,86	0,86
Double Denial (DD)	0,86	0,85	0,86	0,86	0,86	0,86	0,86	0,86
SSW + MLM + SNE	0,87	0,85	0,86	0,86	0,86	0,86	0,86	0,86
SOS + DD	0,87	0,85	0,86	0,86	0,86	0,86	0,86	0,86
SNE + IA	0,87	0,85	0,86	0,86	0,86	0,86	0,86	0,86
MLM + SNE	0,87	0,85	0,86	0,86	0,86	0,86	0,87	0,86
Extending Adjustment	0,87	0,86	0,86	0,86	0,86	0,87	0,87	0,86
MLM + SE	0,87	0,86	0,87	0,87	0,86	0,87	0,87	0,87
BART ProtAugment	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
SE + SNE	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
Keyboard Error	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
SOS + ExtAdj	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
Masked Language Model	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
LAMBADA	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
ExtAdj + DD	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
SE + SNE + IA	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
SSW + SNE	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
SSMBA	0,87	0,86	0,87	0,87	0,87	0,87	0,88	0,87
SSW + SE	0,87	0,86	0,87	0,87	0,87	0,87	0,88	0,87
SSW + MLM + SE + SNE + IA	0,88	0,86	0,87	0,87	0,87	0,87	0,88	0,87
MLM + SNE + IA	0,88	0,86	0,88	0,87	0,87	0,87	0,88	0,87
Synonym Replacement	0,88	0,87	0,88	0,87	0,87	0,87	0,88	0,87
SOS + ExtAdj + DD	0,88	0,87	0,88	0,87	0,87	0,87	0,88	0,87
Random Deletion	0,88	0,87	0,88	0,87	0,87	0,87	0,88	0,87
MLM + SE + IA	0,88	0,87	0,88	0,87	0,87	0,87	0,88	0,88
Insert Adverb (IA)	0,88	0,87	0,88	0,87	0,87	0,88	0,88	0,88
PPDB	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
SSW + MLM + SE	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
Swap Synonym WordNet	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
MLM + IA	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
SSW + SE + SNE	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
SE + IA	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
BF-Translation	0,88	0,87	0,88	0,88	0,88	0,88	0,89	0,88
BERT Augment	0,88	0,88	0,88	0,88	0,88	0,88	0,89	0,88
EDA	0,88	0,88	0,88	0,88	0,88	0,88	0,89	0,88
SSW + MLM	0,89	0,88	0,88	0,88	0,88	0,88	0,89	0,88
Embedding	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
SSW + SNE + IA	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
MLM + SE + SNE	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
SSW + MLM + SE + SNE	0,89	0,89	0,89	0,88	0,88	0,88	0,89	0,88
GAN	0,89	0,89	0,89	0,88	0,88	0,88	0,89	0,88
Deep Backtranslation	0,89	0,89	0,89	0,89	0,88	0,88	0,89	0,88
SSW + IA	0,89	0,89	0,89	0,89	0,88	0,88	0,89	0,88
MLM + SE + SNE + IA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,88
SSW + SE + IA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
SSW + MLM + SNE + IA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
TTA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
Random Synonym Insertion	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
TF-IDF Replacing	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
SSW + SE + SNE + IA	0,89	0,89	0,90	0,89	0,89	0,89	0,89	0,89
SSW + MLM + SE + IA	0,90	0,89	0,90	0,89	0,89	0,89	0,89	0,89
Backtranslation	0,90	0,89	0,90	0,90	0,89	0,90	0,90	0,89
Antonym	0,90	0,89	0,90	0,90	0,90	0,90	0,90	0,89
SSW + MLM + IA	0,90	0,90	0,91	0,91	0,91	0,90	0,90	0,89

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

T5 Parrot	0,90	0,91	0,91	0,91	0,92	0,91	0,91	0,90
Proposed Scheme	0,92	0,91	0,91	0,91	0,92	0,91	0,92	0,92

TABLE 5. The recall values of transformation functions on TRSAv1 dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	BERTurk	DistilBERTurk	ELECTRA	ConvBERTurk
Permutation	0,86	0,85	0,86	0,86	0,86	0,86	0,86	0,86
Swap Named Entity (SNE)	0,86	0,85	0,86	0,86	0,86	0,86	0,87	0,86
No Augmentation	0,86	0,85	0,86	0,87	0,86	0,86	0,87	0,86
Spelling Error (SE)	0,87	0,85	0,86	0,87	0,86	0,86	0,87	0,87
Sentence Order Swap (SOS)	0,87	0,85	0,87	0,87	0,87	0,87	0,87	0,87
Double Denial (DD)	0,87	0,85	0,87	0,87	0,87	0,87	0,87	0,87
SSW + MLM + SNE	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
SOS + DD	0,87	0,86	0,87	0,87	0,87	0,87	0,87	0,87
SNE + IA	0,88	0,86	0,87	0,87	0,87	0,87	0,87	0,87
MLM + SNE	0,88	0,86	0,87	0,87	0,87	0,87	0,87	0,87
Extending Adjustment	0,88	0,86	0,87	0,87	0,87	0,87	0,87	0,87
MLM + SE	0,88	0,86	0,87	0,87	0,87	0,87	0,87	0,87
BART ProtAugment	0,88	0,87	0,88	0,87	0,87	0,87	0,88	0,87
SE + SNE	0,88	0,87	0,88	0,88	0,87	0,87	0,88	0,88
Keyboard Error	0,88	0,87	0,88	0,88	0,87	0,88	0,88	0,88
SOS + ExtAdj	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
Masked Language Model	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
LAMBADA	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
ExtAdj + DD	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
SE + SNE + IA	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
SSW + SNE	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
SSMBA	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
SSW + SE	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
SSW + MLM + SE + SNE + IA	0,88	0,87	0,88	0,88	0,88	0,88	0,88	0,88
MLM + SNE + IA	0,89	0,87	0,88	0,88	0,88	0,88	0,89	0,88
Synonym Replacement	0,89	0,88	0,88	0,88	0,88	0,88	0,89	0,88
SOS + ExtAdj + DD	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
Random Deletion	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
MLM + SE + IA	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
Insert Adverb (IA)	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
PPDB	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,89
SSW + MLM + SE	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,89
Swap Synonym WordNet	0,89	0,88	0,89	0,89	0,89	0,89	0,89	0,89
MLM + IA	0,89	0,88	0,89	0,89	0,89	0,89	0,89	0,89
SSW + SE + SNE	0,89	0,88	0,89	0,89	0,89	0,89	0,89	0,89
SE + IA	0,89	0,88	0,89	0,89	0,89	0,89	0,89	0,89
BF-Translation	0,89	0,88	0,89	0,89	0,89	0,89	0,89	0,89
BERT Augment	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
EDA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
SSW + MLM	0,89	0,89	0,89	0,89	0,89	0,89	0,90	0,89
Embedding	0,89	0,89	0,89	0,89	0,89	0,89	0,90	0,89
SSW + SNE + IA	0,90	0,89	0,89	0,89	0,89	0,89	0,90	0,89
MLM + SE + SNE	0,90	0,89	0,89	0,89	0,89	0,89	0,90	0,89
SSW + MLM + SE + SNE	0,90	0,90	0,89	0,89	0,89	0,89	0,90	0,89
GAN	0,90	0,90	0,89	0,89	0,89	0,89	0,90	0,89
Deep Backtranslation	0,90	0,90	0,90	0,89	0,89	0,89	0,90	0,89
SSW + IA	0,90	0,90	0,90	0,90	0,89	0,89	0,90	0,89
MLM + SE + SNE + IA	0,90	0,90	0,90	0,90	0,89	0,90	0,90	0,89
SSW + SE + IA	0,90	0,90	0,90	0,90	0,89	0,90	0,90	0,90
SSW + MLM + SNE + IA	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
TTA	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

Random Synonym Insertion	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
TF-IDF Replacing	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
SSW + SE + SNE + IA	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
SSW + MLM + SE + IA	0,90	0,90	0,91	0,90	0,90	0,90	0,90	0,90
Backtranslation	0,91	0,90	0,91	0,90	0,90	0,91	0,90	0,90
Antonym	0,91	0,90	0,91	0,91	0,90	0,91	0,91	0,90
SSW + MLM + IA	0,91	0,91	0,92	0,92	0,92	0,91	0,91	0,90
T5 Parrot	0,91	0,92	0,92	0,92	0,93	0,92	0,92	0,91
Proposed Scheme	0,93	0,92	0,92	0,92	0,93	0,92	0,93	0,93

TABLE 6. F-measure values of transformation functions on TRSAv1 dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	BERTurk	DistilBERTurk	ELECTRA	ConvBERTurk
Permutation	0,83	0,83	0,84	0,85	0,84	0,84	0,84	0,83
Swap Named Entity (SNE)	0,85	0,83	0,85	0,86	0,85	0,85	0,85	0,84
No Augmentation	0,85	0,83	0,85	0,86	0,85	0,85	0,85	0,85
Spelling Error (SE)	0,85	0,83	0,85	0,86	0,85	0,85	0,86	0,85
Sentence Order Swap (SOS)	0,86	0,84	0,86	0,86	0,85	0,85	0,86	0,86
Double Denial (DD)	0,86	0,84	0,86	0,86	0,85	0,85	0,86	0,86
SSW + MLM + SNE	0,86	0,84	0,86	0,86	0,86	0,86	0,86	0,86
SOS + DD	0,86	0,84	0,86	0,87	0,86	0,86	0,86	0,86
SNE + IA	0,87	0,84	0,86	0,87	0,86	0,86	0,86	0,86
MLM + SNE	0,87	0,84	0,86	0,87	0,86	0,86	0,86	0,86
Extending Adjustment	0,87	0,84	0,87	0,87	0,86	0,86	0,87	0,86
MLM + SE	0,87	0,85	0,87	0,87	0,86	0,86	0,87	0,86
BART ProtAugment	0,87	0,85	0,87	0,87	0,86	0,86	0,87	0,86
SE + SNE	0,87	0,85	0,87	0,87	0,86	0,86	0,87	0,86
Keyboard Error	0,87	0,85	0,87	0,87	0,87	0,87	0,87	0,86
SOS + ExtAdj	0,87	0,85	0,87	0,87	0,87	0,87	0,87	0,86
Masked Language Model	0,87	0,85	0,87	0,87	0,87	0,87	0,88	0,86
LAMBADA	0,88	0,85	0,87	0,87	0,87	0,87	0,88	0,86
ExtAdj + DD	0,88	0,85	0,87	0,87	0,87	0,87	0,88	0,87
SE + SNE + IA	0,88	0,86	0,87	0,87	0,87	0,87	0,88	0,87
SSW + SNE	0,88	0,86	0,87	0,88	0,87	0,87	0,88	0,87
SSMBA	0,88	0,86	0,87	0,88	0,87	0,87	0,88	0,87
SSW + SE	0,88	0,86	0,87	0,88	0,87	0,87	0,88	0,87
SSW + MLM + SE + SNE + IA	0,88	0,86	0,87	0,88	0,87	0,87	0,88	0,87
MLM + SNE + IA	0,89	0,86	0,88	0,88	0,88	0,87	0,88	0,87
Synonym Replacement	0,89	0,86	0,88	0,88	0,88	0,87	0,88	0,87
SOS + ExtAdj + DD	0,89	0,87	0,88	0,88	0,88	0,87	0,89	0,87
Random Deletion	0,89	0,87	0,88	0,88	0,88	0,87	0,89	0,87
MLM + SE + IA	0,89	0,87	0,88	0,88	0,88	0,87	0,89	0,87
Insert Adverb (IA)	0,89	0,87	0,88	0,88	0,88	0,88	0,89	0,87
PPDB	0,89	0,87	0,89	0,88	0,88	0,88	0,89	0,88
SSW + MLM + SE	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
Swap Synonym WordNet	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
MLM + IA	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
SSW + SE + SNE	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
SE + IA	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
BF-Translation	0,89	0,88	0,89	0,88	0,88	0,88	0,89	0,88
BERT Augment	0,89	0,88	0,89	0,89	0,88	0,88	0,89	0,88
EDA	0,90	0,89	0,89	0,89	0,88	0,89	0,90	0,88
SSW + MLM	0,90	0,89	0,89	0,89	0,88	0,89	0,90	0,88
Embedding	0,90	0,89	0,89	0,89	0,89	0,89	0,90	0,88
SSW + SNE + IA	0,90	0,89	0,90	0,89	0,89	0,89	0,90	0,88
MLM + SE + SNE	0,90	0,89	0,90	0,89	0,89	0,89	0,90	0,88
SSW + MLM + SE + SNE	0,90	0,89	0,90	0,89	0,89	0,89	0,90	0,88
GAN	0,90	0,89	0,90	0,89	0,89	0,89	0,90	0,88
Deep Backtranslation	0,90	0,89	0,90	0,89	0,89	0,89	0,90	0,89

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

SSW + IA	0,90	0,89	0,90	0,89	0,89	0,89	0,90	0,89
MLM + SE + SNE + IA	0,90	0,90	0,90	0,89	0,89	0,89	0,90	0,89
SSW + SE + IA	0,90	0,90	0,90	0,89	0,90	0,89	0,90	0,89
SSW + MLM + SNE + IA	0,91	0,90	0,90	0,90	0,90	0,89	0,90	0,89
TTA	0,91	0,90	0,90	0,90	0,90	0,90	0,90	0,89
Random Synonym Insertion	0,91	0,90	0,90	0,90	0,90	0,90	0,90	0,89
TF-IDF Replacing	0,91	0,90	0,90	0,90	0,90	0,90	0,91	0,89
SSW + SE + SNE + IA	0,91	0,90	0,91	0,90	0,91	0,91	0,91	0,89
SSW + MLM + SE + IA	0,91	0,91	0,91	0,91	0,91	0,91	0,91	0,89
Backtranslation	0,92	0,91	0,92	0,91	0,91	0,91	0,91	0,89
Antonym	0,92	0,91	0,92	0,92	0,91	0,91	0,91	0,90
SSW + MLM + IA	0,92	0,91	0,92	0,92	0,92	0,91	0,92	0,90
T5 Parrot	0,92	0,91	0,92	0,93	0,93	0,92	0,92	0,90
Proposed Scheme	0,93	0,92	0,93	0,93	0,94	0,93	0,93	0,92

TABLE 7. The classification accuracy values of transformation functions on Emotions dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	BERTurk	DistilBERTurk	ELECTRA	ConvBERTurk
Permutation	83,58	84,06	83,97	81,77	84,14	83,22	83,11	82,05
Swap Named Entity (SNE)	84,03	84,20	84,01	83,78	84,41	83,36	84,55	84,41
No Augmentation	84,62	84,23	84,25	84,41	84,56	84,27	84,89	84,67
Spelling Error (SE)	84,74	84,47	84,34	84,73	84,64	84,35	84,98	84,87
Sentence Order Swap (SOS)	85,05	84,71	84,39	84,83	84,72	84,78	85,18	85,16
Double Denial (DD)	85,30	84,72	84,48	84,89	85,74	84,82	85,26	85,40
SSW + MLM + SNE	85,47	84,76	84,93	84,95	85,80	84,82	85,30	85,72
SOS + DD	85,51	84,76	84,98	85,11	85,80	84,83	85,45	85,73
SNE + IA	85,64	84,81	85,03	85,17	85,81	84,89	85,59	85,75
MLM + SNE	85,73	84,83	85,06	85,26	85,96	85,01	85,65	85,82
Extending Adjustment	85,82	85,22	85,09	85,30	86,25	85,01	85,66	85,84
MLM + SE	85,82	85,29	85,18	85,36	86,47	85,36	85,78	85,98
BART ProtAugment	85,84	85,44	85,39	85,41	86,48	85,37	85,79	86,04
SE + SNE	85,87	85,59	85,47	85,60	86,48	86,02	86,01	86,06
Keyboard Error	85,88	85,71	85,53	85,65	86,48	86,09	86,03	86,10
SOS + ExtAdj	86,01	85,97	85,79	85,89	86,48	86,11	86,15	86,16
Masked Language Model	86,04	86,00	85,83	85,92	86,52	86,17	86,18	86,22
LAMBADA	86,10	86,26	85,85	85,99	86,52	86,18	86,19	86,24
ExtAdj + DD	86,17	86,44	85,89	86,01	86,55	86,22	86,26	86,26
SE + SNE + IA	86,23	86,50	85,98	86,13	86,56	86,27	86,30	86,36
SSW + SNE	86,27	86,53	86,02	86,17	86,64	86,37	86,31	86,50
SSMBA	86,43	86,55	86,05	86,20	86,65	86,47	86,31	86,67
SSW + SE	86,43	86,60	86,09	86,28	86,78	86,73	86,39	86,70
SSW + MLM + SE + SNE + IA	86,63	86,61	86,28	86,30	86,82	86,79	86,40	86,74
MLM + SNE + IA	86,68	86,87	86,37	86,37	86,88	86,93	86,45	86,76
Synonym Replacement	86,76	86,88	86,40	86,44	86,92	86,94	86,54	86,79
SOS + ExtAdj + DD	87,03	86,89	86,50	86,46	86,93	87,10	86,55	86,80
Random Deletion	87,11	86,96	86,50	86,46	86,93	87,10	86,59	86,80
MLM + SE + IA	87,11	87,00	86,50	86,57	86,96	87,10	86,65	86,86
Insert Adverb (IA)	87,15	87,08	86,52	86,67	87,12	87,12	86,80	86,87
PPDB	87,16	87,08	86,61	86,68	87,21	87,14	86,82	86,91
SSW + MLM + SE	87,16	87,08	86,79	86,87	87,25	87,14	86,82	86,94
Swap Synonym WordNet	87,19	87,11	86,83	87,01	87,27	87,16	86,85	86,97
MLM + IA	87,29	87,11	86,89	87,12	87,28	87,23	86,99	87,03
SSW + SE + SNE	87,34	87,13	86,90	87,21	87,35	87,27	87,21	87,03
SE + IA	87,36	87,18	86,94	87,36	87,45	87,27	87,25	87,13
BF-Translation	87,37	87,20	87,07	87,37	87,46	87,29	87,28	87,17
BERT Augment	87,48	87,21	87,19	87,48	87,50	87,35	87,34	87,33
EDA	87,57	87,31	87,22	87,48	87,57	87,37	87,40	87,35
SSW + MLM	87,68	87,42	87,23	87,50	87,66	87,39	87,42	87,37

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

Embedding	87,73	87,46	87,48	87,60	87,67	87,47	87,42	87,40
SSW + SNE + IA	87,74	87,66	87,65	87,66	87,79	87,65	87,56	87,52
MLM + SE + SNE	87,88	87,96	87,84	87,73	87,88	87,71	87,57	87,63
SSW + MLM + SE + SNE	87,91	88,03	87,87	87,84	87,89	87,73	87,78	87,73
GAN	87,97	88,09	87,93	87,94	87,99	87,74	87,82	87,73
Deep Backtranslation	88,06	88,12	88,01	88,03	88,04	87,75	87,94	87,84
SSW + IA	88,49	88,25	88,09	88,11	88,12	87,76	88,16	87,88
MLM + SE + SNE + IA	88,50	88,38	88,13	88,12	88,15	87,83	88,16	87,94
SSW + SE + IA	88,53	88,48	88,30	88,51	88,22	88,02	88,36	87,95
SSW + MLM + SNE + IA	88,60	88,74	88,37	88,56	88,33	88,04	88,42	88,02
TTA	88,87	88,92	88,51	88,64	88,48	88,20	88,46	88,05
Random Synonym Insertion	88,95	88,96	88,58	88,66	88,57	88,21	88,55	88,10
TF-IDF Replacing	89,00	88,98	88,65	88,97	88,71	88,44	88,56	88,10
SSW + SE + SNE + IA	89,13	89,08	88,70	89,34	88,77	88,59	88,71	88,60
SSW + MLM + SE + IA	89,21	89,25	88,85	89,35	88,94	88,69	88,77	88,65
Backtranslation	89,22	89,25	88,87	89,48	88,97	88,81	88,82	88,81
Antonym	89,55	89,36	88,91	89,48	88,98	88,98	88,87	88,82
SSW + MLM + IA	89,67	89,84	89,58	89,96	88,99	89,06	89,07	88,97
T5 Parrot	90,01	89,85	89,67	90,00	89,09	89,27	89,30	89,01
Proposed Scheme	90,04	90,41	91,25	90,55	89,51	89,36	90,18	89,89

TABLE 8. The precision values of transformation functions on Emotions dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	BERTurk	DistilBERTurk	ELECTRA	ConvBERTurk
Permutation	0,84	0,85	0,85	0,83	0,85	0,84	0,84	0,83
Swap Named Entity (SNE)	0,85	0,85	0,85	0,85	0,85	0,84	0,85	0,85
No Augmentation	0,85	0,85	0,85	0,85	0,85	0,85	0,86	0,86
Spelling Error (SE)	0,86	0,85	0,85	0,86	0,85	0,85	0,86	0,86
Sentence Order Swap (SOS)	0,86	0,86	0,85	0,86	0,86	0,86	0,86	0,86
Double Denial (DD)	0,86	0,86	0,85	0,86	0,87	0,86	0,86	0,86
SSW + MLM + SNE	0,86	0,86	0,86	0,86	0,87	0,86	0,86	0,87
SOS + DD	0,86	0,86	0,86	0,86	0,87	0,86	0,86	0,87
SNE + IA	0,87	0,86	0,86	0,86	0,87	0,86	0,86	0,87
MLM + SNE	0,87	0,86	0,86	0,86	0,87	0,86	0,87	0,87
Extending Adjustment	0,87	0,86	0,86	0,86	0,87	0,86	0,87	0,87
MLM + SE	0,87	0,86	0,86	0,86	0,87	0,86	0,87	0,87
BART ProtAugment	0,87	0,86	0,86	0,86	0,87	0,86	0,87	0,87
SE + SNE	0,87	0,86	0,86	0,86	0,87	0,87	0,87	0,87
Keyboard Error	0,87	0,87	0,86	0,87	0,87	0,87	0,87	0,87
SOS + ExtAdj	0,87	0,87	0,87	0,87	0,87	0,87	0,87	0,87
Masked Language Model	0,87	0,87	0,87	0,87	0,87	0,87	0,87	0,87
LAMBADA	0,87	0,87	0,87	0,87	0,87	0,87	0,87	0,87
ExtAdj + DD	0,87	0,87	0,87	0,87	0,87	0,87	0,87	0,87
SE + SNE + IA	0,87	0,87	0,87	0,87	0,87	0,87	0,87	0,87
SSW + SNE	0,87	0,87	0,87	0,87	0,88	0,87	0,87	0,87
SSMBA	0,87	0,87	0,87	0,87	0,88	0,87	0,87	0,88
SSW + SE	0,87	0,87	0,87	0,87	0,88	0,88	0,87	0,88
SSW + MLM + SE + SNE + IA	0,88	0,87	0,87	0,87	0,88	0,88	0,87	0,88
MLM + SNE + IA	0,88	0,88	0,87	0,87	0,88	0,88	0,87	0,88
Synonym Replacement	0,88	0,88	0,87	0,87	0,88	0,88	0,87	0,88
SOS + ExtAdj + DD	0,88	0,88	0,87	0,87	0,88	0,88	0,87	0,88
Random Deletion	0,88	0,88	0,87	0,87	0,88	0,88	0,87	0,88
MLM + SE + IA	0,88	0,88	0,87	0,87	0,88	0,88	0,88	0,88
Insert Adverb (IA)	0,88	0,88	0,87	0,88	0,88	0,88	0,88	0,88
PPDB	0,88	0,88	0,87	0,88	0,88	0,88	0,88	0,88
SSW + MLM + SE	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
Swap Synonym WordNet	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
MLM + IA	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

SSW + SE + SNE	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
SE + IA	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
BF-Translation	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
BERT Augment	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
EDA	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
SSW + MLM	0,89	0,88	0,88	0,88	0,89	0,88	0,88	0,88
Embedding	0,89	0,88	0,88	0,88	0,89	0,88	0,88	0,88
SSW + SNE + IA	0,89	0,89	0,89	0,89	0,89	0,89	0,88	0,88
MLM + SE + SNE	0,89	0,89	0,89	0,89	0,89	0,89	0,88	0,89
SSW + MLM + SE + SNE	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
GAN	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
Deep Backtranslation	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
SSW + IA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
MLM + SE + SNE + IA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
SSW + SE + IA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
SSW + MLM + SNE + IA	0,89	0,90	0,89	0,89	0,89	0,89	0,89	0,89
TTA	0,90	0,90	0,89	0,90	0,89	0,89	0,89	0,89
Random Synonym Insertion	0,90	0,90	0,89	0,90	0,89	0,89	0,89	0,89
TF-IDF Replacing	0,90	0,90	0,90	0,90	0,90	0,89	0,89	0,89
SSW + SE + SNE + IA	0,90	0,90	0,90	0,90	0,90	0,89	0,90	0,89
SSW + MLM + SE + IA	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
Backtranslation	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
Antonym	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
SSW + MLM + IA	0,91	0,91	0,90	0,91	0,90	0,90	0,90	0,90
T5 Parrot	0,91	0,91	0,91	0,91	0,90	0,90	0,90	0,90
Proposed Scheme	0,91	0,91	0,92	0,91	0,90	0,90	0,91	0,91

TABLE 9. The recall values of transformation functions on Emotions dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	BERTurk	DistilBERTurk	ELECTRA	ConvBERTurk
Permutation	0,85	0,86	0,86	0,83	0,86	0,85	0,85	0,84
Swap Named Entity (SNE)	0,86	0,86	0,86	0,85	0,86	0,85	0,86	0,86
No Augmentation	0,86	0,86	0,86	0,86	0,86	0,86	0,87	0,86
Spelling Error (SE)	0,86	0,86	0,86	0,86	0,86	0,86	0,87	0,87
Sentence Order Swap (SOS)	0,87	0,86	0,86	0,87	0,86	0,87	0,87	0,87
Double Denial (DD)	0,87	0,86	0,86	0,87	0,87	0,87	0,87	0,87
SSW + MLM + SNE	0,87	0,86	0,87	0,87	0,88	0,87	0,87	0,87
SOS + DD	0,87	0,86	0,87	0,87	0,88	0,87	0,87	0,87
SNE + IA	0,87	0,87	0,87	0,87	0,88	0,87	0,87	0,87
MLM + SNE	0,87	0,87	0,87	0,87	0,88	0,87	0,87	0,88
Extending Adjustment	0,88	0,87	0,87	0,87	0,88	0,87	0,87	0,88
MLM + SE	0,88	0,87	0,87	0,87	0,88	0,87	0,88	0,88
BART ProtAugment	0,88	0,87	0,87	0,87	0,88	0,87	0,88	0,88
SE + SNE	0,88	0,87	0,87	0,87	0,88	0,88	0,88	0,88
Keyboard Error	0,88	0,87	0,87	0,87	0,88	0,88	0,88	0,88
SOS + ExtAdj	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
Masked Language Model	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
LAMBADA	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
ExtAdj + DD	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
SE + SNE + IA	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
SSW + SNE	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
SSMBA	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
SSW + SE	0,88	0,88	0,88	0,88	0,89	0,88	0,88	0,88
SSW + MLM + SE + SNE + IA	0,88	0,88	0,88	0,88	0,89	0,89	0,88	0,89
MLM + SNE + IA	0,88	0,89	0,88	0,88	0,89	0,89	0,88	0,89
Synonym Replacement	0,89	0,89	0,88	0,88	0,89	0,89	0,88	0,89
SOS + ExtAdj + DD	0,89	0,89	0,88	0,88	0,89	0,89	0,88	0,89
Random Deletion	0,89	0,89	0,88	0,88	0,89	0,89	0,88	0,89

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

MLM + SE + IA	0,89	0,89	0,88	0,88	0,89	0,89	0,88	0,89
Insert Adverb (IA)	0,89	0,89	0,88	0,88	0,89	0,89	0,89	0,89
PPDB	0,89	0,89	0,88	0,88	0,89	0,89	0,89	0,89
SSW + MLM + SE	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
Swap Synonym WordNet	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
MLM + IA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
SSW + SE + SNE	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
SE + IA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
BF-Translation	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
BERT Augment	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
EDA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
SSW + MLM	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
Embedding	0,90	0,89	0,89	0,89	0,89	0,89	0,89	0,89
SSW + SNE + IA	0,90	0,89	0,89	0,89	0,90	0,89	0,89	0,89
MLM + SE + SNE	0,90	0,90	0,90	0,90	0,90	0,90	0,89	0,89
SSW + MLM + SE + SNE	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
GAN	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
Deep Backtranslation	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
SSW + IA	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
MLM + SE + SNE + IA	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
SSW + SE + IA	0,90	0,90	0,90	0,90	0,90	0,90	0,90	0,90
SSW + MLM + SNE + IA	0,90	0,91	0,90	0,90	0,90	0,90	0,90	0,90
TTA	0,91	0,91	0,90	0,90	0,90	0,90	0,90	0,90
Random Synonym Insertion	0,91	0,91	0,90	0,90	0,90	0,90	0,90	0,90
TF-IDF Replacing	0,91	0,91	0,90	0,91	0,91	0,90	0,90	0,90
SSW + SE + SNE + IA	0,91	0,91	0,91	0,91	0,91	0,90	0,91	0,90
SSW + MLM + SE + IA	0,91	0,91	0,91	0,91	0,91	0,90	0,91	0,90
Backtranslation	0,91	0,91	0,91	0,91	0,91	0,91	0,91	0,91
Antonym	0,91	0,91	0,91	0,91	0,91	0,91	0,91	0,91
SSW + MLM + IA	0,91	0,92	0,91	0,92	0,91	0,91	0,91	0,91
T5 Parrot	0,92	0,92	0,92	0,92	0,91	0,91	0,91	0,91
Proposed Scheme	0,92	0,92	0,93	0,92	0,91	0,91	0,92	0,92

TABLE 10. The F-measure values of transformation functions on Emotions dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	BERTurk	DistilBERTurk	ELECTRA	ConvBERTurk
Permutation	0,85	0,85	0,86	0,82	0,86	0,85	0,85	0,82
Swap Named Entity (SNE)	0,85	0,86	0,85	0,85	0,86	0,84	0,86	0,85
No Augmentation	0,86	0,86	0,85	0,86	0,86	0,85	0,86	0,86
Spelling Error (SE)	0,86	0,86	0,85	0,86	0,86	0,85	0,86	0,86
Sentence Order Swap (SOS)	0,87	0,86	0,85	0,86	0,86	0,86	0,86	0,86
Double Denial (DD)	0,87	0,86	0,86	0,86	0,88	0,86	0,86	0,86
SSW + MLM + SNE	0,87	0,86	0,86	0,86	0,88	0,86	0,86	0,87
SOS + DD	0,87	0,86	0,86	0,86	0,88	0,86	0,87	0,87
SNE + IA	0,87	0,86	0,86	0,86	0,88	0,86	0,87	0,87
MLM + SNE	0,87	0,86	0,86	0,86	0,88	0,86	0,87	0,87
Extending Adjustment	0,87	0,87	0,86	0,86	0,88	0,86	0,87	0,87
MLM + SE	0,87	0,87	0,86	0,86	0,88	0,86	0,87	0,87
BART ProtAugment	0,87	0,87	0,87	0,86	0,88	0,86	0,87	0,87
SE + SNE	0,87	0,87	0,87	0,86	0,88	0,87	0,87	0,87
Keyboard Error	0,87	0,87	0,87	0,87	0,88	0,87	0,87	0,87
SOS + ExtAdj	0,87	0,87	0,87	0,87	0,88	0,87	0,87	0,87
Masked Language Model	0,87	0,87	0,87	0,87	0,88	0,87	0,87	0,87
LAMBADA	0,87	0,88	0,87	0,87	0,88	0,87	0,87	0,87
ExtAdj + DD	0,87	0,88	0,87	0,87	0,88	0,88	0,88	0,87
SE + SNE + IA	0,87	0,88	0,87	0,87	0,88	0,88	0,88	0,87
SSW + SNE	0,87	0,88	0,87	0,87	0,88	0,88	0,88	0,87
SSMBA	0,88	0,88	0,87	0,87	0,88	0,88	0,87	0,88

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

SSW + SE	0,88	0,88	0,87	0,87	0,88	0,88	0,88	0,88
SSW + MLM + SE + SNE + IA	0,88	0,88	0,88	0,87	0,88	0,88	0,88	0,88
MLM + SNE + IA	0,88	0,88	0,88	0,87	0,88	0,89	0,88	0,88
Synonym Replacement	0,88	0,88	0,88	0,88	0,88	0,88	0,88	0,88
SOS + ExtAdj + DD	0,88	0,88	0,88	0,88	0,88	0,89	0,88	0,88
Random Deletion	0,89	0,89	0,88	0,88	0,88	0,89	0,88	0,88
MLM + SE + IA	0,88	0,89	0,88	0,88	0,88	0,89	0,88	0,88
Insert Adverb (IA)	0,89	0,89	0,88	0,88	0,88	0,89	0,88	0,88
PPDB	0,89	0,89	0,88	0,88	0,89	0,89	0,88	0,88
SSW + MLM + SE	0,89	0,89	0,88	0,88	0,89	0,89	0,88	0,88
Swap Synonym WordNet	0,89	0,89	0,88	0,88	0,89	0,89	0,88	0,88
MLM + IA	0,89	0,89	0,88	0,88	0,89	0,89	0,88	0,88
SSW + SE + SNE	0,89	0,89	0,88	0,88	0,89	0,89	0,89	0,88
SE + IA	0,89	0,89	0,88	0,89	0,89	0,89	0,89	0,88
BF-Translation	0,89	0,89	0,88	0,89	0,89	0,89	0,89	0,88
BERT Augment	0,89	0,89	0,88	0,89	0,89	0,89	0,89	0,88
EDA	0,89	0,89	0,88	0,89	0,89	0,89	0,89	0,88
SSW + MLM	0,89	0,89	0,88	0,89	0,89	0,89	0,89	0,88
Embedding	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,88
SSW + SNE + IA	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,88
MLM + SE + SNE	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,88
SSW + MLM + SE + SNE	0,89	0,89	0,89	0,89	0,89	0,89	0,89	0,89
GAN	0,89	0,90	0,89	0,89	0,89	0,89	0,89	0,89
Deep Backtranslation	0,89	0,90	0,89	0,89	0,90	0,89	0,89	0,89
SSW + IA	0,90	0,90	0,89	0,89	0,90	0,89	0,90	0,88
MLM + SE + SNE + IA	0,90	0,90	0,89	0,89	0,90	0,89	0,90	0,89
SSW + SE + IA	0,90	0,90	0,90	0,90	0,90	0,89	0,90	0,89
SSW + MLM + SNE + IA	0,90	0,90	0,90	0,90	0,90	0,89	0,90	0,89
TTA	0,90	0,90	0,90	0,90	0,90	0,89	0,90	0,89
Random Synonym Insertion	0,90	0,91	0,90	0,90	0,90	0,89	0,90	0,89
TF-IDF Replacing	0,90	0,91	0,90	0,90	0,90	0,90	0,90	0,89
SSW + SE + SNE + IA	0,91	0,91	0,90	0,91	0,90	0,90	0,90	0,89
SSW + MLM + SE + IA	0,91	0,91	0,90	0,91	0,90	0,90	0,90	0,89
Backtranslation	0,91	0,91	0,90	0,91	0,90	0,90	0,90	0,89
Antonym	0,91	0,91	0,90	0,91	0,90	0,90	0,90	0,89
SSW + MLM + IA	0,91	0,91	0,91	0,92	0,90	0,90	0,90	0,90
T5 Parrot	0,91	0,91	0,91	0,92	0,90	0,91	0,91	0,89
Proposed Scheme	0,91	0,91	0,93	0,92	0,91	0,90	0,92	0,91

TABLE 11. The classification accuracy values of transformation functions on MR dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	ELECTRA
Permutation	87,70	88,01	87,10	86,59	86,91
Swap Named Entity (SNE)	87,81	88,28	88,93	87,72	87,98
No Augmentation	88,02	88,30	89,04	88,33	88,22
Spelling Error (SE)	88,35	88,35	89,11	88,45	88,86
Sentence Order Swap (SOS)	88,37	88,48	89,25	88,49	89,13
Double Denial (DD)	88,63	89,05	89,31	88,60	89,16
SSW + MLM + SNE	88,72	89,05	89,37	88,66	89,32
SOS + DD	88,93	89,11	89,38	88,91	89,50
SNE + IA	89,01	89,53	89,55	88,93	89,56
MLM + SNE	89,40	89,64	89,69	89,06	89,63
Extending Adjustment	89,52	89,68	89,77	89,16	89,90
MLM + SE	89,55	89,73	89,87	89,32	89,91
BART ProtAugment	89,63	89,84	89,98	89,48	89,93
SE + SNE	89,66	89,93	90,02	89,53	89,99
Keyboard Error	89,85	89,98	90,10	89,70	90,09
SOS + ExtAdj	89,87	90,02	90,13	89,74	90,18

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

Masked Language Model	89,98	90,09	90,43	89,75	90,23
LAMBADA	90,07	90,15	90,48	89,75	90,25
ExtAdj + DD	90,20	90,23	90,69	90,00	90,30
SE + SNE + IA	90,21	90,24	90,77	90,05	90,34
SSW + SNE	90,24	90,27	90,83	90,06	90,41
SSMBA	90,40	90,31	90,92	90,10	90,43
SSW + SE	90,44	90,34	90,94	90,16	90,60
SSW + MLM + SE + SNE + IA	90,50	90,35	90,95	90,19	90,68
MLM + SNE + IA	90,50	90,50	90,96	90,28	90,81
Synonym Replacement	90,59	90,52	91,02	90,30	90,81
SOS + ExtAdj + DD	90,63	90,59	91,04	90,38	90,84
Random Deletion	90,71	90,59	91,07	90,43	90,94
MLM + SE + IA	90,74	90,60	91,10	90,70	91,03
Insert Adverb (IA)	90,75	90,60	91,16	90,80	91,07
PPDB	90,77	90,62	91,21	90,83	91,17
SSW + MLM + SE	90,83	90,63	91,31	91,05	91,30
Swap Synonym WordNet	90,92	90,71	91,42	91,09	91,32
MLM + IA	91,08	90,76	91,45	91,10	91,47
SSW + SE + SNE	91,09	90,89	91,61	91,31	91,51
SE + IA	91,20	90,90	91,81	91,44	91,55
BF-Translation	91,27	90,93	91,85	91,47	91,61
BERT Augment	91,40	90,98	91,88	91,48	91,66
EDA	91,41	91,08	92,04	91,51	91,68
SSW + MLM	91,45	91,12	92,07	91,62	91,68
Embedding	91,80	91,15	92,13	91,77	91,69
SSW + SNE + IA	91,96	91,23	92,18	91,82	91,79
MLM + SE + SNE	92,01	91,23	92,35	91,86	91,86
SSW + MLM + SE + SNE	92,32	91,36	92,68	91,99	91,91
GAN	92,33	91,50	92,71	92,02	91,92
Deep Backtranslation	92,33	91,68	92,82	92,13	91,92
SSW + IA	92,64	91,71	92,85	92,17	91,93
MLM + SE + SNE + IA	92,78	91,76	92,91	92,26	92,03
SSW + SE + IA	92,92	91,77	92,95	92,27	92,08
SSW + MLM + SNE + IA	92,97	91,82	92,98	92,33	92,12
TTA	93,08	91,90	93,05	92,43	92,21
Random Synonym Insertion	93,08	92,12	93,14	92,53	92,21
TF-IDF Replacing	93,25	92,26	93,42	92,60	92,28
SSW + SE + SNE + IA	93,30	92,63	93,48	92,70	92,38
SSW + MLM + SE + IA	93,36	92,69	93,57	93,43	92,47
Backtranslation	93,46	92,87	93,63	93,51	92,74
Antonym	93,57	92,98	93,66	93,60	93,00
SSW + MLM + IA	93,85	93,19	93,80	93,73	93,02
T5 Parrot	93,86	93,38	94,03	94,26	93,47
Proposed Scheme	93,96	93,42	94,41	94,44	94,12

TABLE 12. The precision values of transformation functions on MR dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	ELECTRA
Permutation	0,89	0,89	0,88	0,87	0,88
Swap Named Entity (SNE)	0,89	0,89	0,90	0,89	0,89
No Augmentation	0,89	0,89	0,90	0,89	0,89
Spelling Error (SE)	0,89	0,89	0,90	0,89	0,90
Sentence Order Swap (SOS)	0,89	0,89	0,90	0,89	0,90
Double Denial (DD)	0,90	0,90	0,90	0,89	0,90
SSW + MLM + SNE	0,90	0,90	0,90	0,90	0,90
SOS + DD	0,90	0,90	0,90	0,90	0,90
SNE + IA	0,90	0,90	0,90	0,90	0,90
MLM + SNE	0,90	0,91	0,91	0,90	0,91
Extending Adjustment	0,90	0,91	0,91	0,90	0,91

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

MLM + SE	0,90	0,91	0,91	0,90	0,91
BART ProtAugment	0,91	0,91	0,91	0,90	0,91
SE + SNE	0,91	0,91	0,91	0,90	0,91
Keyboard Error	0,91	0,91	0,91	0,91	0,91
SOS + ExtAdj	0,91	0,91	0,91	0,91	0,91
Masked Language Model	0,91	0,91	0,91	0,91	0,91
LAMBADA	0,91	0,91	0,91	0,91	0,91
ExtAdj + DD	0,91	0,91	0,92	0,91	0,91
SE + SNE + IA	0,91	0,91	0,92	0,91	0,91
SSW + SNE	0,91	0,91	0,92	0,91	0,91
SSMBA	0,91	0,91	0,92	0,91	0,91
SSW + SE	0,91	0,91	0,92	0,91	0,92
SSW + MLM + SE + SNE + IA	0,91	0,91	0,92	0,91	0,92
MLM + SNE + IA	0,91	0,91	0,92	0,91	0,92
Synonym Replacement	0,92	0,91	0,92	0,91	0,92
SOS + ExtAdj + DD	0,92	0,92	0,92	0,91	0,92
Random Deletion	0,92	0,92	0,92	0,91	0,92
MLM + SE + IA	0,92	0,92	0,92	0,92	0,92
Insert Adverb (IA)	0,92	0,92	0,92	0,92	0,92
PPDB	0,92	0,92	0,92	0,92	0,92
SSW + MLM + SE	0,92	0,92	0,92	0,92	0,92
Swap Synonym WordNet	0,92	0,92	0,92	0,92	0,92
MLM + IA	0,92	0,92	0,92	0,92	0,92
SSW + SE + SNE	0,92	0,92	0,93	0,92	0,92
SE + IA	0,92	0,92	0,93	0,92	0,92
BF-Translation	0,92	0,92	0,93	0,92	0,93
BERT Augment	0,92	0,92	0,93	0,92	0,93
EDA	0,92	0,92	0,93	0,92	0,93
SSW + MLM	0,92	0,92	0,93	0,93	0,93
Embedding	0,93	0,92	0,93	0,93	0,93
SSW + SNE + IA	0,93	0,92	0,93	0,93	0,93
MLM + SE + SNE	0,93	0,92	0,93	0,93	0,93
SSW + MLM + SE + SNE	0,93	0,92	0,94	0,93	0,93
GAN	0,93	0,92	0,94	0,93	0,93
Deep Backtranslation	0,93	0,93	0,94	0,93	0,93
SSW + IA	0,94	0,93	0,94	0,93	0,93
MLM + SE + SNE + IA	0,94	0,93	0,94	0,93	0,93
SSW + SE + IA	0,94	0,93	0,94	0,93	0,93
SSW + MLM + SNE + IA	0,94	0,93	0,94	0,93	0,93
TTA	0,94	0,93	0,94	0,93	0,93
Random Synonym Insertion	0,94	0,93	0,94	0,93	0,93
TF-IDF Replacing	0,94	0,93	0,94	0,94	0,93
SSW + SE + SNE + IA	0,94	0,94	0,94	0,94	0,93
SSW + MLM + SE + IA	0,94	0,94	0,95	0,94	0,93
Backtranslation	0,94	0,94	0,95	0,94	0,94
Antonym	0,95	0,94	0,95	0,95	0,94
SSW + MLM + IA	0,95	0,94	0,95	0,95	0,94
T5 Parrot	0,95	0,94	0,95	0,95	0,94
Proposed Scheme	0,95	0,94	0,95	0,95	0,95

TABLE 13. The recall values of transformation functions on MR dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	ELECTRA
Permutation	0,89	0,90	0,89	0,88	0,89
Swap Named Entity (SNE)	0,90	0,90	0,91	0,90	0,90
No Augmentation	0,90	0,90	0,91	0,90	0,90
Spelling Error (SE)	0,90	0,90	0,91	0,90	0,91
Sentence Order Swap (SOS)	0,90	0,90	0,91	0,90	0,91

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

Double Denial (DD)	0,90	0,91	0,91	0,90	0,91
SSW + MLM + SNE	0,91	0,91	0,91	0,90	0,91
SOS + DD	0,91	0,91	0,91	0,91	0,91
SNE + IA	0,91	0,91	0,91	0,91	0,91
MLM + SNE	0,91	0,91	0,92	0,91	0,91
Extending Adjustment	0,91	0,92	0,92	0,91	0,92
MLM + SE	0,91	0,92	0,92	0,91	0,92
BART ProtAugment	0,91	0,92	0,92	0,91	0,92
SE + SNE	0,91	0,92	0,92	0,91	0,92
Keyboard Error	0,92	0,92	0,92	0,92	0,92
SOS + ExtAdj	0,92	0,92	0,92	0,92	0,92
Masked Language Model	0,92	0,92	0,92	0,92	0,92
LAMBADA	0,92	0,92	0,92	0,92	0,92
ExtAdj + DD	0,92	0,92	0,93	0,92	0,92
SE + SNE + IA	0,92	0,92	0,93	0,92	0,92
SSW + SNE	0,92	0,92	0,93	0,92	0,92
SSMBA	0,92	0,92	0,93	0,92	0,92
SSW + SE	0,92	0,92	0,93	0,92	0,92
SSW + MLM + SE + SNE + IA	0,92	0,92	0,93	0,92	0,93
MLM + SNE + IA	0,92	0,92	0,93	0,92	0,93
Synonym Replacement	0,92	0,92	0,93	0,92	0,93
SOS + ExtAdj + DD	0,92	0,92	0,93	0,92	0,93
Random Deletion	0,93	0,92	0,93	0,92	0,93
MLM + SE + IA	0,93	0,92	0,93	0,93	0,93
Insert Adverb (IA)	0,93	0,92	0,93	0,93	0,93
PPDB	0,93	0,92	0,93	0,93	0,93
SSW + MLM + SE	0,93	0,92	0,93	0,93	0,93
Swap Synonym WordNet	0,93	0,93	0,93	0,93	0,93
MLM + IA	0,93	0,93	0,93	0,93	0,93
SSW + SE + SNE	0,93	0,93	0,93	0,93	0,93
SE + IA	0,93	0,93	0,94	0,93	0,93
BF-Translation	0,93	0,93	0,94	0,93	0,93
BERT Augment	0,93	0,93	0,94	0,93	0,94
EDA	0,93	0,93	0,94	0,93	0,94
SSW + MLM	0,93	0,93	0,94	0,93	0,94
Embedding	0,94	0,93	0,94	0,94	0,94
SSW + SNE + IA	0,94	0,93	0,94	0,94	0,94
MLM + SE + SNE	0,94	0,93	0,94	0,94	0,94
SSW + MLM + SE + SNE	0,94	0,93	0,95	0,94	0,94
GAN	0,94	0,93	0,95	0,94	0,94
Deep Backtranslation	0,94	0,94	0,95	0,94	0,94
SSW + IA	0,95	0,94	0,95	0,94	0,94
MLM + SE + SNE + IA	0,95	0,94	0,95	0,94	0,94
SSW + SE + IA	0,95	0,94	0,95	0,94	0,94
SSW + MLM + SNE + IA	0,95	0,94	0,95	0,94	0,94
TTA	0,95	0,94	0,95	0,94	0,94
Random Synonym Insertion	0,95	0,94	0,95	0,94	0,94
TF-IDF Replacing	0,95	0,94	0,95	0,94	0,94
SSW + SE + SNE + IA	0,95	0,95	0,95	0,95	0,94
SSW + MLM + SE + IA	0,95	0,95	0,95	0,95	0,94
Backtranslation	0,95	0,95	0,96	0,95	0,95
Antonym	0,95	0,95	0,96	0,96	0,95
SSW + MLM + IA	0,96	0,95	0,96	0,96	0,95
T5 Parrot	0,96	0,95	0,96	0,96	0,95
Proposed Scheme	0,96	0,95	0,96	0,96	0,96

TABLE 14. The F-measure values of transformation functions on MR dataset.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	ELECTRA
Permutation	0,89	0,90	0,89	0,88	0,88
Swap Named Entity (SNE)	0,89	0,89	0,91	0,89	0,89
No Augmentation	0,89	0,89	0,91	0,90	0,89
Spelling Error (SE)	0,90	0,89	0,91	0,90	0,90
Sentence Order Swap (SOS)	0,90	0,89	0,91	0,90	0,91
Double Denial (DD)	0,90	0,90	0,91	0,90	0,91
SSW + MLM + SNE	0,90	0,90	0,91	0,90	0,91
SOS + DD	0,90	0,90	0,91	0,90	0,91
SNE + IA	0,90	0,91	0,91	0,90	0,91
MLM + SNE	0,91	0,91	0,91	0,90	0,91
Extending Adjustment	0,91	0,91	0,91	0,91	0,91
MLM + SE	0,91	0,91	0,92	0,91	0,91
BART ProtAugment	0,91	0,91	0,92	0,91	0,91
SE + SNE	0,91	0,91	0,92	0,91	0,91
Keyboard Error	0,91	0,91	0,92	0,91	0,91
SOS + ExtAdj	0,91	0,91	0,92	0,91	0,92
Masked Language Model	0,91	0,91	0,92	0,91	0,92
LAMBADA	0,91	0,91	0,92	0,91	0,92
ExtAdj + DD	0,92	0,91	0,92	0,91	0,92
SE + SNE + IA	0,92	0,91	0,93	0,91	0,92
SSW + SNE	0,92	0,91	0,93	0,91	0,92
SSMBA	0,92	0,91	0,93	0,91	0,92
SSW + SE	0,92	0,91	0,93	0,92	0,92
SSW + MLM + SE + SNE + IA	0,92	0,91	0,93	0,92	0,92
MLM + SNE + IA	0,92	0,92	0,93	0,92	0,92
Synonym Replacement	0,92	0,92	0,93	0,92	0,92
SOS + ExtAdj + DD	0,92	0,92	0,93	0,92	0,92
Random Deletion	0,92	0,92	0,93	0,92	0,92
MLM + SE + IA	0,92	0,92	0,93	0,92	0,92
Insert Adverb (IA)	0,92	0,92	0,93	0,92	0,92
PPDB	0,92	0,92	0,93	0,92	0,93
SSW + MLM + SE	0,92	0,92	0,93	0,92	0,93
Swap Synonym WordNet	0,92	0,92	0,93	0,92	0,93
MLM + IA	0,93	0,92	0,93	0,92	0,93
SSW + SE + SNE	0,93	0,92	0,93	0,93	0,93
SE + IA	0,93	0,92	0,93	0,93	0,93
BF-Translation	0,93	0,92	0,93	0,93	0,93
BERT Augment	0,93	0,92	0,93	0,93	0,93
EDA	0,93	0,92	0,94	0,93	0,93
SSW + MLM	0,93	0,92	0,94	0,93	0,93
Embedding	0,94	0,92	0,94	0,93	0,93
SSW + SNE + IA	0,94	0,92	0,94	0,93	0,93
MLM + SE + SNE	0,94	0,92	0,94	0,93	0,93
SSW + MLM + SE + SNE	0,94	0,92	0,94	0,94	0,93
GAN	0,94	0,92	0,94	0,94	0,93
Deep Backtranslation	0,94	0,93	0,95	0,94	0,93
SSW + IA	0,95	0,93	0,95	0,94	0,93
MLM + SE + SNE + IA	0,95	0,93	0,95	0,94	0,93
SSW + SE + IA	0,95	0,93	0,95	0,94	0,93
SSW + MLM + SNE + IA	0,95	0,93	0,95	0,94	0,93
TTA	0,95	0,93	0,95	0,94	0,94
Random Synonym Insertion	0,95	0,93	0,95	0,94	0,93
TF-IDF Replacing	0,95	0,93	0,95	0,94	0,93
SSW + SE + SNE + IA	0,95	0,94	0,95	0,94	0,94
SSW + MLM + SE + IA	0,95	0,94	0,95	0,95	0,94
Backtranslation	0,95	0,94	0,95	0,95	0,94
Antonym	0,95	0,94	0,95	0,95	0,94
SSW + MLM + IA	0,96	0,94	0,95	0,95	0,94

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

T5 Parrot	0,96	0,94	0,95	0,96	0,95
Proposed Scheme	0,96	0,94	0,96	0,96	0,96

TABLE 15. The classification accuracy values of transformation functions on Swahili News dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	ELECTRA
Permutation	0,73	0,72	0,73	0,78	0,73
Swap Named Entity (SNE)	0,73	0,74	0,74	0,75	0,73
No Augmentation	0,74	0,74	0,74	0,73	0,73
Spelling Error (SE)	0,74	0,74	0,74	0,77	0,73
Sentence Order Swap (SOS)	0,74	0,75	0,75	0,77	0,74
Double Denial (DD)	0,74	0,75	0,75	0,78	0,74
SSW + MLM + SNE	0,74	0,75	0,75	0,77	0,74
SOS + DD	0,75	0,75	0,75	0,75	0,75
SNE + IA	0,75	0,75	0,75	0,77	0,75
MLM + SNE	0,75	0,75	0,75	0,76	0,75
Extending Adjustment	0,75	0,75	0,75	0,76	0,75
MLM + SE	0,75	0,75	0,75	0,75	0,75
BART ProtAugment	0,75	0,75	0,75	0,77	0,75
SE + SNE	0,75	0,76	0,75	0,77	0,75
Keyboard Error	0,76	0,76	0,75	0,76	0,75
SOS + ExtAdj	0,76	0,76	0,76	0,76	0,75
Masked Language Model	0,76	0,76	0,76	0,76	0,75
LAMBADA	0,76	0,76	0,76	0,78	0,76
ExtAdj + DD	0,76	0,76	0,76	0,79	0,76
SE + SNE + IA	0,76	0,76	0,76	0,79	0,76
SSW + SNE	0,76	0,76	0,76	0,74	0,76
SSMBA	0,76	0,76	0,76	0,77	0,76
SSW + SE	0,76	0,76	0,76	0,79	0,76
SSW + MLM + SE + SNE + IA	0,76	0,76	0,76	0,77	0,76
MLM + SNE + IA	0,76	0,77	0,76	0,76	0,76
Synonym Replacement	0,76	0,77	0,76	0,77	0,76
SOS + ExtAdj + DD	0,76	0,77	0,76	0,78	0,76
Random Deletion	0,77	0,77	0,77	0,77	0,76
MLM + SE + IA	0,77	0,77	0,77	0,80	0,76
Insert Adverb (IA)	0,77	0,77	0,77	0,77	0,76
PPDB	0,77	0,77	0,77	0,78	0,76
SSW + MLM + SE	0,77	0,77	0,77	0,74	0,77
Swap Synonym WordNet	0,77	0,77	0,77	0,74	0,77
MLM + IA	0,77	0,77	0,77	0,76	0,77
SSW + SE + SNE	0,77	0,77	0,77	0,78	0,77
SE + IA	0,77	0,77	0,77	0,77	0,77
BF-Translation	0,77	0,77	0,77	0,77	0,77
BERT Augment	0,77	0,77	0,77	0,77	0,77
EDA	0,77	0,77	0,77	0,78	0,77
SSW + MLM	0,77	0,77	0,77	0,80	0,77
Embedding	0,78	0,77	0,77	0,79	0,77
SSW + SNE + IA	0,78	0,78	0,77	0,78	0,77
MLM + SE + SNE	0,78	0,78	0,78	0,76	0,77
SSW + MLM + SE + SNE	0,78	0,78	0,78	0,75	0,77
GAN	0,78	0,78	0,78	0,77	0,77
Deep Backtranslation	0,78	0,78	0,78	0,78	0,77
SSW + IA	0,78	0,78	0,78	0,76	0,78
MLM + SE + SNE + IA	0,78	0,78	0,78	0,78	0,78
SSW + SE + IA	0,78	0,78	0,78	0,76	0,78
SSW + MLM + SNE + IA	0,78	0,78	0,78	0,77	0,78
TTA	0,78	0,78	0,78	0,77	0,78
Random Synonym Insertion	0,79	0,78	0,78	0,75	0,78
TF-IDF Replacing	0,79	0,79	0,78	0,77	0,78
SSW + SE + SNE + IA	0,79	0,79	0,79	0,77	0,78
SSW + MLM + SE + IA	0,79	0,79	0,79	0,78	0,79

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

Backtranslation	0,80	0,79	0,79	0,76	0,79
Antonym	0,80	0,80	0,79	0,77	0,79
SSW + MLM + IA	0,80	0,80	0,79	0,75	0,79
T5 Parrot	0,80	0,81	0,79	0,76	0,79
Proposed Scheme	0,81	0,81	0,80	0,76	0,80

TABLE 16. The precision values of transformation functions on Swahili News dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	ELECTRA
Permutation	0,73	0,72	0,73	0,78	0,73
Swap Named Entity (SNE)	0,73	0,74	0,74	0,75	0,73
No Augmentation	0,74	0,74	0,74	0,73	0,73
Spelling Error (SE)	0,74	0,74	0,74	0,77	0,73
Sentence Order Swap (SOS)	0,74	0,75	0,75	0,77	0,74
Double Denial (DD)	0,74	0,75	0,75	0,78	0,74
SSW + MLM + SNE	0,74	0,75	0,75	0,77	0,74
SOS + DD	0,75	0,75	0,75	0,75	0,75
SNE + IA	0,75	0,75	0,75	0,77	0,75
MLM + SNE	0,75	0,75	0,75	0,76	0,75
Extending Adjustment	0,75	0,75	0,75	0,76	0,75
MLM + SE	0,75	0,75	0,75	0,75	0,75
BART ProtAugment	0,75	0,75	0,75	0,77	0,75
SE + SNE	0,75	0,76	0,75	0,77	0,75
Keyboard Error	0,76	0,76	0,75	0,76	0,75
SOS + ExtAdj	0,76	0,76	0,76	0,76	0,75
Masked Language Model	0,76	0,76	0,76	0,76	0,75
LAMBADA	0,76	0,76	0,76	0,78	0,76
ExtAdj + DD	0,76	0,76	0,76	0,79	0,76
SE + SNE + IA	0,76	0,76	0,76	0,79	0,76
SSW + SNE	0,76	0,76	0,76	0,74	0,76
SSMBA	0,76	0,76	0,76	0,77	0,76
SSW + SE	0,76	0,76	0,76	0,79	0,76
SSW + MLM + SE + SNE + IA	0,76	0,76	0,76	0,77	0,76
MLM + SNE + IA	0,76	0,77	0,76	0,76	0,76
Synonym Replacement	0,76	0,77	0,76	0,77	0,76
SOS + ExtAdj + DD	0,76	0,77	0,76	0,78	0,76
Random Deletion	0,77	0,77	0,77	0,77	0,76
MLM + SE + IA	0,77	0,77	0,77	0,80	0,76
Insert Adverb (IA)	0,77	0,77	0,77	0,77	0,76
PPDB	0,77	0,77	0,77	0,78	0,76
SSW + MLM + SE	0,77	0,77	0,77	0,74	0,77
Swap Synonym WordNet	0,77	0,77	0,77	0,74	0,77
MLM + IA	0,77	0,77	0,77	0,76	0,77
SSW + SE + SNE	0,77	0,77	0,77	0,78	0,77
SE + IA	0,77	0,77	0,77	0,77	0,77
BF-Translation	0,77	0,77	0,77	0,77	0,77
BERT Augment	0,77	0,77	0,77	0,77	0,77
EDA	0,77	0,77	0,77	0,78	0,77
SSW + MLM	0,77	0,77	0,77	0,80	0,77
Embedding	0,78	0,77	0,77	0,79	0,77
SSW + SNE + IA	0,78	0,78	0,77	0,78	0,77
MLM + SE + SNE	0,78	0,78	0,78	0,76	0,77
SSW + MLM + SE + SNE	0,78	0,78	0,78	0,75	0,77
GAN	0,78	0,78	0,78	0,77	0,77
Deep Backtranslation	0,78	0,78	0,78	0,78	0,77
SSW + IA	0,78	0,78	0,78	0,76	0,78
MLM + SE + SNE + IA	0,78	0,78	0,78	0,78	0,78

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

SSW + SE + IA	0,78	0,78	0,78	0,76	0,78
SSW + MLM + SNE + IA	0,78	0,78	0,78	0,77	0,78
TTA	0,78	0,78	0,78	0,77	0,78
Random Synonym Insertion	0,79	0,78	0,78	0,75	0,78
TF-IDF Replacing	0,79	0,79	0,78	0,77	0,78
SSW + SE + SNE + IA	0,79	0,79	0,79	0,77	0,78
SSW + MLM + SE + IA	0,79	0,79	0,79	0,78	0,79
Backtranslation	0,80	0,79	0,79	0,76	0,79
Antonym	0,80	0,80	0,79	0,77	0,79
SSW + MLM + IA	0,80	0,80	0,79	0,75	0,79
T5 Parrot	0,80	0,81	0,79	0,76	0,79
Proposed Scheme	0,81	0,81	0,80	0,76	0,80

TABLE 17. The recall values of transformation functions on Swahili News dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	ELECTRA
Permutation	0,74	0,73	0,74	0,79	0,74
Swap Named Entity (SNE)	0,74	0,74	0,74	0,76	0,74
No Augmentation	0,74	0,75	0,75	0,73	0,74
Spelling Error (SE)	0,75	0,75	0,75	0,78	0,74
Sentence Order Swap (SOS)	0,75	0,75	0,75	0,78	0,75
Double Denial (DD)	0,75	0,75	0,76	0,78	0,75
SSW + MLM + SNE	0,75	0,76	0,76	0,78	0,75
SOS + DD	0,76	0,76	0,76	0,76	0,75
SNE + IA	0,76	0,76	0,76	0,78	0,76
MLM + SNE	0,76	0,76	0,76	0,77	0,76
Extending Adjustment	0,76	0,76	0,76	0,76	0,76
MLM + SE	0,76	0,76	0,76	0,76	0,76
BART ProtAugment	0,76	0,76	0,76	0,78	0,76
SE + SNE	0,76	0,77	0,76	0,78	0,76
Keyboard Error	0,76	0,77	0,76	0,77	0,76
SOS + ExtAdj	0,76	0,77	0,76	0,77	0,76
Masked Language Model	0,76	0,77	0,77	0,77	0,76
LAMBADA	0,77	0,77	0,77	0,79	0,76
ExtAdj + DD	0,77	0,77	0,77	0,79	0,76
SE + SNE + IA	0,77	0,77	0,77	0,79	0,76
SSW + SNE	0,77	0,77	0,77	0,75	0,76
SSMBA	0,77	0,77	0,77	0,78	0,77
SSW + SE	0,77	0,77	0,77	0,80	0,77
SSW + MLM + SE + SNE + IA	0,77	0,77	0,77	0,78	0,77
MLM + SNE + IA	0,77	0,77	0,77	0,76	0,77
Synonym Replacement	0,77	0,78	0,77	0,78	0,77
SOS + ExtAdj + DD	0,77	0,78	0,77	0,79	0,77
Random Deletion	0,77	0,78	0,77	0,78	0,77
MLM + SE + IA	0,77	0,78	0,77	0,81	0,77
Insert Adverb (IA)	0,77	0,78	0,77	0,78	0,77
PPDB	0,78	0,78	0,77	0,79	0,77
SSW + MLM + SE	0,78	0,78	0,78	0,75	0,77
Swap Synonym WordNet	0,78	0,78	0,78	0,74	0,77
MLM + IA	0,78	0,78	0,78	0,77	0,78
SSW + SE + SNE	0,78	0,78	0,78	0,79	0,78
SE + IA	0,78	0,78	0,78	0,78	0,78
BF-Translation	0,78	0,78	0,78	0,78	0,78
BERT Augment	0,78	0,78	0,78	0,78	0,78
EDA	0,78	0,78	0,78	0,79	0,78
SSW + MLM	0,78	0,78	0,78	0,81	0,78
Embedding	0,78	0,78	0,78	0,79	0,78
SSW + SNE + IA	0,78	0,78	0,78	0,78	0,78

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

MLM + SE + SNE	0,79	0,78	0,78	0,76	0,78
SSW + MLM + SE + SNE	0,79	0,79	0,78	0,76	0,78
GAN	0,79	0,79	0,79	0,77	0,78
Deep Backtranslation	0,79	0,79	0,79	0,78	0,78
SSW + IA	0,79	0,79	0,79	0,77	0,79
MLM + SE + SNE + IA	0,79	0,79	0,79	0,78	0,79
SSW + SE + IA	0,79	0,79	0,79	0,76	0,79
SSW + MLM + SNE + IA	0,79	0,79	0,79	0,78	0,79
TTA	0,79	0,79	0,79	0,78	0,79
Random Synonym Insertion	0,79	0,79	0,79	0,76	0,79
TF-IDF Replacing	0,79	0,79	0,79	0,77	0,79
SSW + SE + SNE + IA	0,80	0,80	0,80	0,78	0,79
SSW + MLM + SE + IA	0,80	0,80	0,80	0,79	0,79
Backtranslation	0,81	0,80	0,80	0,77	0,80
Antonym	0,81	0,81	0,80	0,78	0,80
SSW + MLM + IA	0,81	0,81	0,80	0,76	0,80
T5 Parrot	0,81	0,81	0,80	0,76	0,80
Proposed Scheme	0,81	0,82	0,81	0,77	0,81

TABLE 18. The F-measure values of transformation functions on Swahili News dataset.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	ELECTRA
Permutation	0,73	0,72	0,72	0,81	0,73
Swap Named Entity (SNE)	0,73	0,74	0,73	0,76	0,73
No Augmentation	0,74	0,74	0,75	0,72	0,74
Spelling Error (SE)	0,74	0,75	0,73	0,79	0,73
Sentence Order Swap (SOS)	0,74	0,75	0,74	0,79	0,75
Double Denial (DD)	0,74	0,75	0,74	0,79	0,75
SSW + MLM + SNE	0,74	0,75	0,74	0,78	0,75
SOS + DD	0,75	0,75	0,75	0,76	0,75
SNE + IA	0,75	0,75	0,74	0,79	0,75
MLM + SNE	0,76	0,75	0,75	0,77	0,75
Extending Adjustment	0,75	0,76	0,75	0,76	0,76
MLM + SE	0,76	0,76	0,76	0,76	0,76
BART ProtAugment	0,75	0,76	0,75	0,78	0,76
SE + SNE	0,76	0,76	0,75	0,78	0,76
Keyboard Error	0,76	0,76	0,76	0,76	0,76
SOS + ExtAdj	0,76	0,76	0,76	0,77	0,76
Masked Language Model	0,76	0,76	0,76	0,77	0,76
LAMBADA	0,76	0,77	0,75	0,79	0,76
ExtAdj + DD	0,76	0,77	0,75	0,80	0,76
SE + SNE + IA	0,76	0,77	0,75	0,80	0,76
SSW + SNE	0,76	0,77	0,78	0,73	0,76
SSMBA	0,76	0,77	0,76	0,78	0,76
SSW + SE	0,76	0,77	0,75	0,81	0,76
SSW + MLM + SE + SNE + IA	0,77	0,77	0,76	0,78	0,76
MLM + SNE + IA	0,77	0,77	0,77	0,75	0,76
Synonym Replacement	0,77	0,77	0,76	0,78	0,76
SOS + ExtAdj + DD	0,76	0,77	0,76	0,79	0,76
Random Deletion	0,77	0,77	0,77	0,78	0,76
MLM + SE + IA	0,77	0,77	0,75	0,82	0,76
Insert Adverb (IA)	0,77	0,77	0,77	0,78	0,76
PPDB	0,77	0,78	0,76	0,79	0,77
SSW + MLM + SE	0,77	0,78	0,78	0,74	0,77
Swap Synonym WordNet	0,77	0,78	0,79	0,72	0,77
MLM + IA	0,77	0,78	0,78	0,76	0,77
SSW + SE + SNE	0,77	0,78	0,77	0,79	0,77
SE + IA	0,77	0,78	0,77	0,77	0,77

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

BF-Translation	0,77	0,78	0,77	0,78	0,77
BERT Augment	0,78	0,78	0,77	0,78	0,77
EDA	0,78	0,78	0,77	0,79	0,77
SSW + MLM	0,78	0,78	0,76	0,81	0,78
Embedding	0,78	0,78	0,77	0,79	0,78
SSW + SNE + IA	0,78	0,78	0,77	0,78	0,78
MLM + SE + SNE	0,78	0,78	0,79	0,75	0,78
SSW + MLM + SE + SNE	0,78	0,78	0,79	0,74	0,78
GAN	0,78	0,78	0,79	0,76	0,78
Deep Backtranslation	0,79	0,78	0,79	0,78	0,78
SSW + IA	0,79	0,78	0,79	0,76	0,78
MLM + SE + SNE + IA	0,79	0,78	0,79	0,77	0,78
SSW + SE + IA	0,79	0,78	0,80	0,74	0,78
SSW + MLM + SNE + IA	0,79	0,78	0,79	0,77	0,78
TTA	0,79	0,78	0,79	0,77	0,78
Random Synonym Insertion	0,79	0,79	0,81	0,73	0,78
TF-IDF Replacing	0,79	0,79	0,80	0,76	0,78
SSW + SE + SNE + IA	0,79	0,79	0,80	0,77	0,79
SSW + MLM + SE + IA	0,79	0,79	0,80	0,78	0,79
Backtranslation	0,81	0,79	0,81	0,75	0,79
Antonym	0,81	0,80	0,80	0,76	0,79
SSW + MLM + IA	0,81	0,81	0,82	0,74	0,79
T5 Parrot	0,80	0,82	0,82	0,74	0,79
Proposed Scheme	0,81	0,82	0,82	0,75	0,80

TABLE 19. The BLUE scores of text augmentation models.

Transformation Function	TRSAv1	Emotions	MR	Swahili News
Permutation	40,21	40,30	40,27	39,97
Swap Named Entity (SNE)	40,18	40,30	40,24	39,91
Spelling Error (SE)	40,17	40,23	40,23	39,88
Sentence Order Swap (SOS)	40,09	40,17	40,23	39,84
Double Denial (DD)	40,07	40,14	40,23	39,82
SSW + MLM + SNE	40,05	40,12	40,22	39,72
SOS + DD	40,02	40,01	40,13	39,71
SNE + IA	39,92	39,89	40,13	39,70
MLM + SNE	39,78	39,89	40,12	39,66
Extending Adjustment	39,76	39,85	40,07	39,65
MLM + SE	39,69	39,69	39,89	39,65
BART ProtAugment	39,65	39,52	39,88	39,58
SE + SNE	39,55	39,36	39,83	39,56
Keyboard Error	39,45	39,30	39,81	39,54
SOS + ExtAdj	39,26	39,22	39,79	39,53
Masked Language Model	39,12	38,96	39,72	39,35
LAMBADA	39,10	38,93	39,53	39,23
ExtAdj + DD	39,09	38,89	39,33	39,14
SE + SNE + IA	38,95	38,69	39,23	39,12
SSW + SNE	38,90	38,67	39,21	38,93
SSMBA	38,79	38,39	39,13	38,89
SSW + SE	38,79	38,20	39,12	38,84
SSW + MLM + SE + SNE + IA	38,77	38,06	39,10	38,84
MLM + SNE + IA	38,71	38,01	39,08	38,84
Synonym Replacement	38,67	37,92	39,06	38,78

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

SOS + ExtAdj + DD	38,47	37,83	38,93	38,62
Random Deletion	38,45	37,82	38,91	38,49
MLM + SE + IA	38,34	37,81	38,88	38,41
Insert Adverb (IA)	38,27	37,75	38,75	38,33
PPDB	38,23	37,72	38,60	38,29
SSW + MLM + SE	38,07	37,64	38,55	38,21
Swap Synonym WordNet	37,92	37,45	38,50	38,19
MLM + IA	37,87	37,32	38,35	38,18
SSW + SE + SNE	37,70	37,29	38,29	38,17
SE + IA	37,52	37,21	38,17	38,12
BF-Translation	37,12	37,01	38,12	37,92
BERT Augment	37,03	36,85	38,10	37,91
EDA	36,93	36,85	37,91	37,75
SSW + MLM	36,81	36,31	37,79	37,71
Embedding	36,76	36,08	37,47	37,32
SSW + SNE + IA	36,63	35,98	37,38	37,07
MLM + SE + SNE	36,59	35,77	37,23	36,51
SSW + MLM + SE + SNE	36,49	35,75	37,00	36,50
GAN	36,48	35,48	36,83	36,35
Deep Backtranslation	36,37	35,47	36,76	35,90
SSW + IA	36,30	35,30	36,73	35,89
MLM + SE + SNE + IA	36,17	34,99	36,50	35,73
SSW + SE + IA	35,80	34,71	36,48	35,73
SSW + MLM + SNE + IA	35,66	34,45	36,47	35,69
TTA	35,53	34,36	36,10	35,45
Random Synonym Insertion	35,46	33,46	36,05	35,42
TF-IDF Replacing	35,19	33,39	35,98	35,33
SSW + SE + SNE + IA	35,18	33,11	35,89	35,23
SSW + MLM + SE + IA	35,10	33,10	35,55	34,57
Backtranslation	34,76	32,63	35,45	34,39
Antonym	34,36	32,47	34,70	34,04
SSW + MLM + IA	34,29	31,76	34,22	33,31
T5 Parrot	33,63	31,55	32,77	31,51
Proposed Scheme	31,33	29,14	30,82	30,68

TABLE 20. The Cosine similarity scores of text augmentation models.

Transformation Function	TRSAv1	Emotions	MR	Swahili News
Permutation	0,79	0,79	0,80	0,80
Swap Named Entity (SNE)	0,79	0,79	0,80	0,80
Spelling Error (SE)	0,79	0,79	0,80	0,80
Sentence Order Swap (SOS)	0,79	0,79	0,80	0,80
Double Denial (DD)	0,80	0,80	0,80	0,80
SSW + MLM + SNE	0,80	0,80	0,80	0,80
SOS + DD	0,80	0,80	0,80	0,80
SNE + IA	0,80	0,80	0,80	0,80
MLM + SNE	0,80	0,80	0,80	0,80
Extending Adjustment	0,80	0,80	0,80	0,80
MLM + SE	0,80	0,80	0,80	0,80
BART ProtAugment	0,80	0,80	0,80	0,80

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

SE + SNE	0,80	0,80	0,80	0,80
Keyboard Error	0,81	0,80	0,80	0,80
SOS + ExtAdj	0,81	0,80	0,80	0,80
Masked Language Model	0,81	0,80	0,80	0,80
LAMBADA	0,81	0,80	0,80	0,80
ExtAdj + DD	0,81	0,81	0,81	0,80
SE + SNE + IA	0,81	0,81	0,81	0,81
SSW + SNE	0,81	0,81	0,81	0,81
SSMBA	0,81	0,81	0,81	0,81
SSW + SE	0,81	0,81	0,81	0,81
SSW + MLM + SE + SNE + IA	0,81	0,81	0,81	0,81
MLM + SNE + IA	0,81	0,81	0,81	0,81
Synonym Replacement	0,81	0,81	0,81	0,81
SOS + ExtAdj + DD	0,82	0,81	0,81	0,81
Random Deletion	0,82	0,81	0,81	0,81
MLM + SE + IA	0,82	0,81	0,81	0,81
Insert Adverb (IA)	0,82	0,82	0,81	0,81
PPDB	0,82	0,82	0,82	0,81
SSW + MLM + SE	0,82	0,82	0,82	0,81
Swap Synonym WordNet	0,82	0,82	0,82	0,82
MLM + IA	0,82	0,82	0,82	0,82
SSW + SE + SNE	0,82	0,82	0,82	0,82
SE + IA	0,82	0,82	0,82	0,82
BF-Translation	0,82	0,82	0,82	0,82
BERT Augment	0,82	0,82	0,82	0,82
EDA	0,83	0,82	0,82	0,82
SSW + MLM	0,83	0,82	0,82	0,82
Embedding	0,83	0,83	0,83	0,82
SSW + SNE + IA	0,83	0,83	0,83	0,83
MLM + SE + SNE	0,83	0,83	0,83	0,83
SSW + MLM + SE + SNE	0,84	0,83	0,83	0,83
GAN	0,84	0,83	0,83	0,83
Deep Backtranslation	0,84	0,83	0,83	0,83
SSW + IA	0,84	0,83	0,83	0,83
MLM + SE + SNE + IA	0,84	0,84	0,83	0,83
SSW + SE + IA	0,84	0,84	0,83	0,83
SSW + MLM + SNE + IA	0,85	0,84	0,83	0,83
TTA	0,85	0,84	0,84	0,84
Random Synonym Insertion	0,85	0,84	0,84	0,84
TF-IDF Replacing	0,85	0,84	0,84	0,84
SSW + SE + SNE + IA	0,85	0,84	0,85	0,84
SSW + MLM + SE + IA	0,86	0,85	0,85	0,85
Backtranslation	0,86	0,85	0,86	0,85
Antonym	0,86	0,85	0,86	0,85
SSW + MLM + IA	0,87	0,86	0,86	0,86
T5 Parrot	0,89	0,86	0,87	0,87
Proposed Scheme	0,93	0,87	0,88	0,89

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

In Figures 2-4, the main effects plot for classification accuracy, precision, and recall, have been given, respectively. Regarding the performance of deep learning models, they yield similar predictive performance. The choice of the augmentation percentage can significantly impact the performance of text augmentation techniques. In our experiments, we have evaluated different augmentation percentages, ranging from 10% to 100% of the training data size. The empirical results suggest that a 40% augmentation percentage yields the highest performance. The best BLUE scores and cosine similarity values have been also achieved for this configuration, as illustrated by Figures 5-6. This outcome can be explained by several factors:

Balancing Act: Augmenting too little data (e.g., 10% or 20%) may not introduce enough diversity into the training set to have a substantial impact on model performance. On the other hand, augmenting too much data (e.g., 80%, 90%, or 100%) may dilute the original training data, making it harder for the model to learn from the core examples. A 40% augmentation strikes a balance between introducing diversity and preserving the core training examples.

Overfitting vs. Generalization: When you augment a small percentage of data, the risk of overfitting to the original examples remains high. Conversely, when you augment a large percentage of data, the model may struggle to generalize to unseen examples. A 40% augmentation rate appears to strike a sweet spot, where the model benefits from added diversity while retaining the ability to generalize to new instances effectively.

Data Quality: The quality of the augmented data matters. If the augmentation techniques introduce noisy or irrelevant examples, it can degrade model performance. A 40% augmentation rate may allow for effective data expansion while still maintaining high-quality augmented samples.

Computational Resources: Augmenting a large portion of the data (e.g., 80% or 100%) can be computationally expensive, especially if the augmentation techniques are complex or require significant processing power. A 40% augmentation rate is more computationally efficient while still providing substantial benefits.

Dataset Specifics: The optimal augmentation percentage can also depend on the specific dataset, its size, and the inherent complexity of the text classification task. Some datasets may require more augmentation to compensate for limited training data, while others may not benefit as much from extensive augmentation.

The choice of a 40% augmentation rate likely represents a well-balanced trade-off between introducing diversity, preserving data quality, and efficient resource utilization. However, it's essential to note that the optimal percentage may vary depending on the specific dataset and task, so experimentation and fine-tuning of this parameter may be necessary for different scenarios.

To summarize the empirical results, we present the performance comparison of text augmentation models in terms of BLUE scores and cosine similarity scores in Figure 7 and Figure 8, respectively.

In Table 21, we present the results of the ANOVA tests conducted to assess the statistical significance of various factors affecting the performance of our text classification models. The factors under investigation include the dataset used, the specific text transformation function applied, and the choice of deep learning model architecture. These tests are crucial for identifying which factors significantly influence the dependent variable (in this case, model performance) and for quantifying the extent of their impact. The ANOVA test for the dataset factor yielded a significant F-value of 89331.47 ($p < 0.001$), indicating that the choice of dataset has a highly significant effect on model performance. In practical terms, this means that different datasets used for training and evaluation led to significantly different model outcomes. Post-hoc tests, such as Tukey's Honestly Significant Difference (HSD), can be employed to further investigate pairwise differences between specific datasets. The transformation function factor also exhibited a highly significant impact on model performance, as evidenced by an F-value of 297.60 ($p < 0.001$). This result suggests that the various text transformation techniques employed significantly influenced the performance of the models. Post-hoc tests can be used to identify specific transformation functions that led to statistically different model performances. The choice of deep learning model architecture was found to be statistically significant with an F-value of 10.05 ($p < 0.001$). This indicates that different model architectures had a significant impact on model performance. Further analysis, such as pairwise comparisons, can help elucidate which specific model architectures led to significantly different results. The error term, representing unexplained variability in model performance, had a relatively low mean square value of 0.00002, indicating that the models' performance was consistent across individual runs within each factor level. The total variance was partitioned among the factors and the error term. The results of the ANOVA tests highlight the importance of careful consideration of dataset selection, text transformation techniques, and deep learning model architecture when developing text classification models. These factors are not only statistically significant but also exert varying degrees of influence on model performance.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

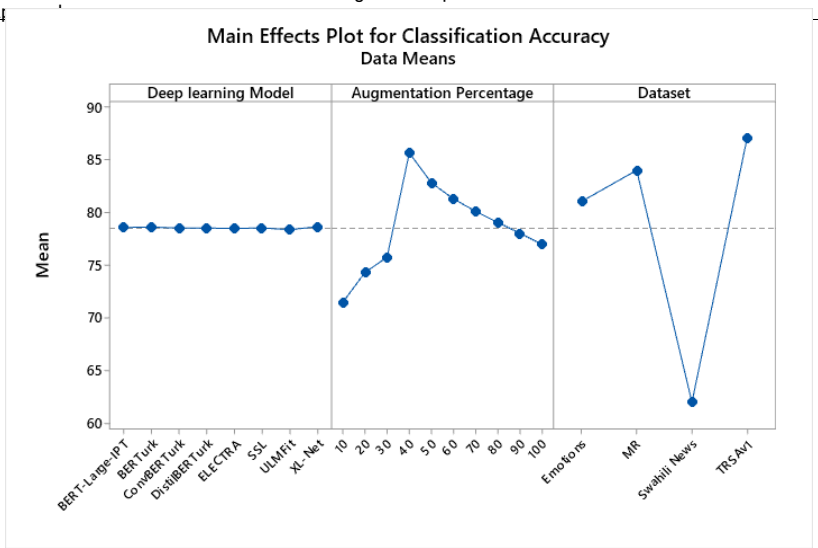


FIGURE 2. The main effects plot for classification accuracy.

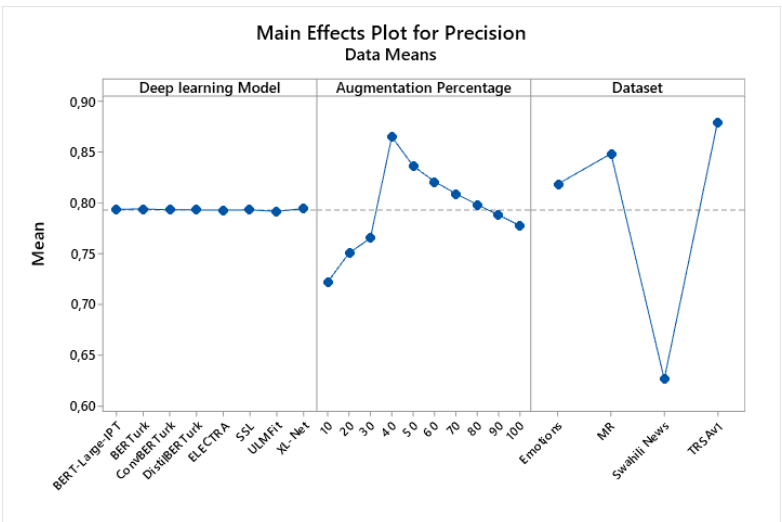


FIGURE 3. The main effects plot for precision.

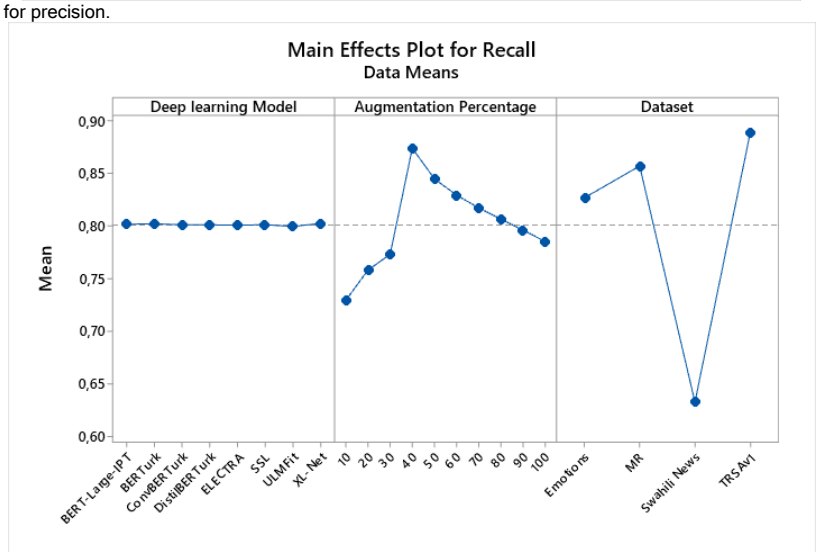


FIGURE 4. The main effects plot for recall.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

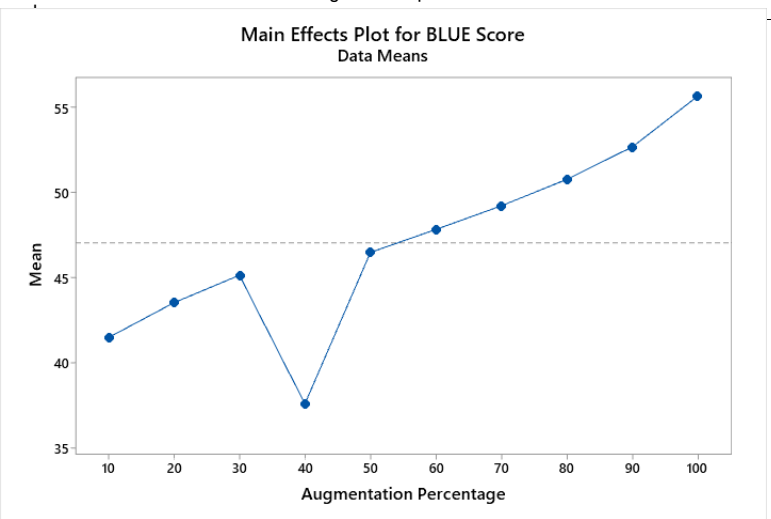


FIGURE 5. The main effects plot for BLUE Score based on augmentation percentage.

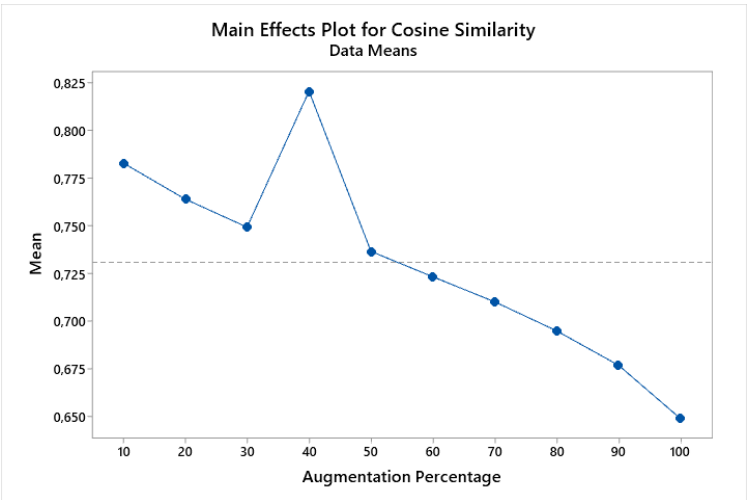


FIGURE 6. The main effects plot for Cosine similarity based on augmentation percentage.

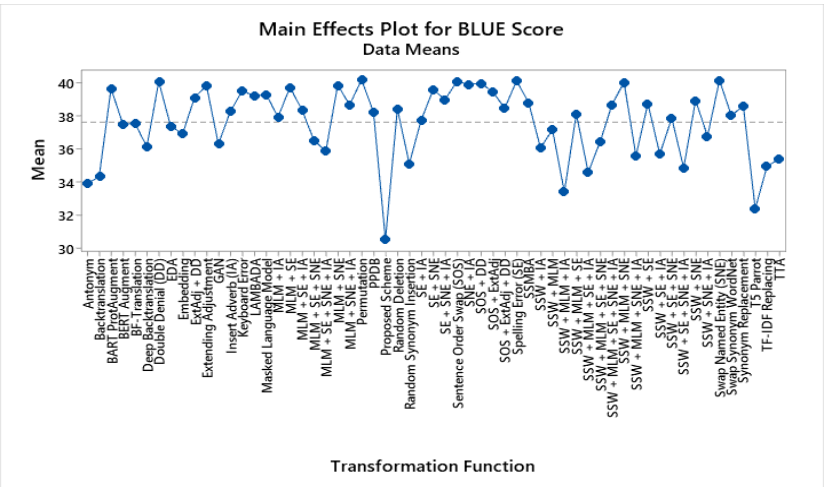


FIGURE 7. The main effects plot for BLUE Score based on transformation functions.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

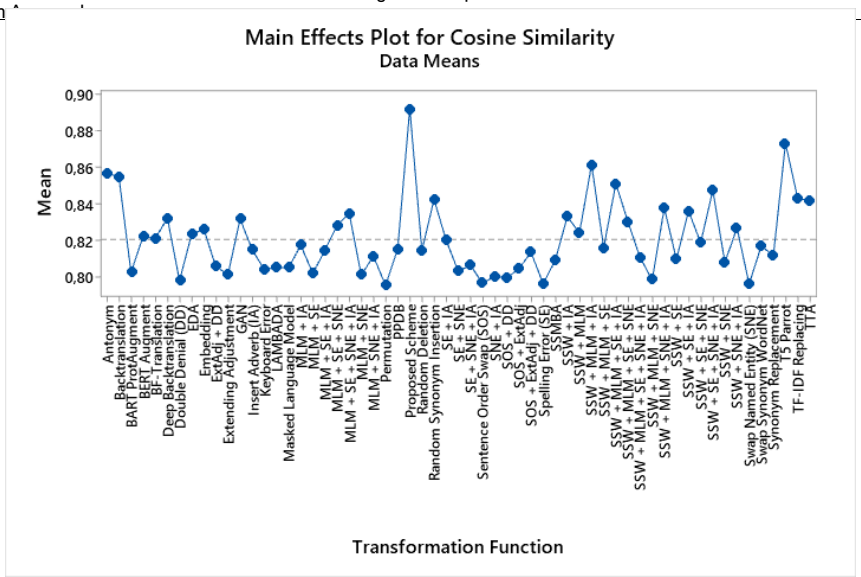


FIGURE 8. The main effects plot for cosine similarity based on transformation functions.

TABLE 21. The ANOVA test results of empirical analysis.

Source	DF	Adj SS	Adj MS	F-Value	P-Value
Dataset	3	6,13105	2,04368	89331,47	0,000
Transformation Function	59	0,40170	0,00681	297,60	0,000
Deep Learning Model	7	0,00161	0,00023	10,05	0,000
Error	1850	0,04232	0,00002		
Total	1919	6,57667			

TABLE 22. The empirical results in terms of training times of the models in hours.

Transformation Function	XL-Net	ULM-Fit	BERT-Large-ITPT	SSL	BERTurk	DistilBERTurk	ELECTRA	ConvBERTurk
Permutation	10,88	5,17	3,73	7,63	4,32	9,93	7,38	6,85
Swap Named Entity (SNE)	7,13	6,17	10,01	6,32	6,48	7,41	7,64	7,20
No Augmentation	0,58	1,99	1,24	2,13	1,03	3,12	1,87	2,77
Spelling Error (SE)	2,48	2,30	1,60	3,10	1,53	3,46	2,60	3,06
Sentence Order Swap (SOS)	2,81	2,59	2,50	3,27	2,13	3,88	2,94	3,67
Double Denial (DD)	3,07	2,61	2,97	3,34	2,54	4,30	3,16	3,77
SSW + MLM + SNE	3,20	2,80	3,00	3,65	2,56	4,87	3,94	4,36
SOS + DD	3,21	3,59	3,10	3,89	3,39	5,07	4,08	4,50
SNE + IA	3,23	3,61	3,30	4,19	3,52	5,38	4,15	5,04
MLM + SNE	3,69	3,77	3,91	4,19	5,01	5,53	4,30	5,11
Extending Adjustment	3,95	4,14	3,98	4,31	5,08	5,62	4,49	5,14
MLM + SE	4,47	4,20	4,03	4,76	5,36	5,69	4,99	5,15
BART ProtAugment	4,57	4,27	4,04	4,78	5,54	5,70	5,04	5,38
SE + SNE	4,59	4,28	4,13	4,93	5,65	5,83	5,09	5,46
Keyboard Error	4,65	4,32	4,26	4,96	5,77	5,89	5,12	5,47
SOS + ExtAdj	4,79	4,34	4,27	4,98	6,01	5,91	5,34	5,47
Masked Language Model	4,87	4,76	4,48	5,03	6,04	6,02	5,41	5,57
LAMBADA	5,06	4,79	4,51	5,22	6,08	6,07	5,50	5,62
ExtAdj + DD	5,10	4,82	4,63	5,25	6,11	6,16	5,72	5,67
SE + SNE + IA	5,10	4,93	4,96	5,26	6,26	6,17	5,74	5,69

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

SSW + SNE	5,35	5,09	5,11	5,50	6,32	6,17	5,76	5,77
SSMBA	5,58	5,10	5,18	5,61	6,38	6,26	5,93	5,77
SSW + SE	5,77	5,33	5,29	5,62	6,43	6,34	6,00	5,90
SSW + MLM + SE + SNE + IA	5,93	5,54	5,36	5,65	6,46	6,41	6,02	6,05
MLM + SNE + IA	5,95	5,57	5,51	5,67	6,47	6,41	6,09	6,06
Synonym Replacement	6,00	5,93	5,52	5,73	6,47	6,54	6,11	6,29
SOS + ExtAdj + DD	6,14	5,93	5,66	5,83	6,58	6,54	6,12	6,32
Random Deletion	6,18	5,98	5,80	5,94	6,68	6,57	6,12	6,47
MLM + SE + IA	6,18	6,08	5,86	6,36	6,71	6,57	6,13	6,50
Insert Adverb (IA)	6,41	6,26	6,24	6,37	6,95	6,60	6,23	6,69
PPDB	6,46	6,46	6,45	6,80	7,01	6,61	6,31	6,85
SSW + MLM + SE	6,58	6,55	6,46	6,90	7,36	6,75	6,38	6,99
Swap Synonym WordNet	6,59	7,07	6,49	6,95	7,49	6,93	6,42	7,37
MLM + IA	6,63	7,08	6,59	6,97	7,49	7,17	6,66	7,56
SSW + SE + SNE	6,64	7,19	6,60	7,00	7,55	7,21	6,70	7,80
SE + IA	6,71	7,30	6,62	7,03	7,70	7,25	6,75	7,86
BF-Translation	6,87	7,36	6,98	7,07	7,82	7,31	6,93	8,07
BERT Augment	6,99	7,50	7,15	7,39	7,84	7,56	7,05	8,14
EDA	7,05	7,53	7,20	7,53	7,88	7,71	7,19	8,15
SSW + MLM	7,33	7,58	7,32	8,07	7,92	7,72	7,27	8,15
Embedding	7,47	7,67	7,37	8,09	7,99	7,83	7,55	8,20
SSW + SNE + IA	7,53	7,72	7,45	8,11	8,43	8,12	7,57	8,24
MLM + SE + SNE	7,60	7,88	8,14	8,16	8,45	8,19	7,60	8,64
SSW + MLM + SE + SNE	7,88	8,11	8,27	8,29	8,54	8,34	7,69	8,67
GAN	7,90	8,16	8,28	8,54	8,63	8,37	7,83	9,06
Deep Backtranslation	7,92	8,41	8,41	8,55	8,71	8,61	7,84	9,10
SSW + IA	8,24	8,49	8,42	8,61	8,97	8,63	7,91	9,13
MLM + SE + SNE + IA	8,51	8,64	8,71	8,79	9,04	9,00	7,97	9,40
SSW + SE + IA	9,34	8,70	8,72	8,85	9,14	9,25	8,20	9,56
SSW + MLM + SNE + IA	9,38	8,77	8,79	9,00	9,17	9,31	8,31	9,58
TTA	9,38	9,13	9,29	9,40	9,27	9,32	8,39	9,66
Random Synonym Insertion	9,71	9,17	9,30	9,46	9,40	9,53	8,48	9,72
TF-IDF Replacing	9,75	9,30	9,58	9,56	9,41	9,77	9,08	9,84
SSW + SE + SNE + IA	9,80	9,46	9,93	9,93	10,00	9,82	9,64	9,90
SSW + MLM + SE + IA	9,84	9,88	10,02	9,98	10,04	9,95	9,74	10,58
Backtranslation	10,45	10,44	10,17	10,01	10,13	10,00	9,94	10,59
Antonym	10,93	10,99	10,21	10,21	10,89	10,02	10,01	11,07
SSW + MLM + IA	11,63	11,02	11,19	10,29	12,21	10,09	10,08	11,31
T5 Parrot	11,72	12,12	11,85	10,37	12,63	10,12	10,09	12,20
Proposed Scheme	14,14	12,27	12,52	12,71	12,90	10,49	12,09	13,21

To assess the efficiency and effectiveness of the augmentation techniques and models, we conducted experiments on a benchmark dataset using various augmentation strategies. The results, shown in Table 22, provide insights into the training times and performance metrics associated with each combination of model and augmentation technique. From Table 22, it is evident that there exists a substantial variation in training times and performance metrics across different augmentation techniques and models. Notably, the choice of augmentation technique significantly influences both training efficiency and model performance. Augmentation strategies such as "No Augmentation" and "Spelling Error (SE)" tend to result in lower training times, while more complex techniques like "Proposed Scheme" lead to longer training durations. Moreover, we observe that certain models exhibit varying degrees of sensitivity to the augmentation techniques. For instance, ConvBERTurk tends to demonstrate relatively higher training times across the board, while ELECTRA shows competitive performance across multiple augmentation

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

strategies. These findings emphasize the importance of carefully selecting augmentation techniques in conjunction with the choice of model to strike a balance between training efficiency and model performance. Researchers and practitioners should consider the specific requirements of their tasks when making these selections.

V. DISCUSSION

The Turkish language, with its unique linguistic characteristics, poses both challenges and opportunities in the field of sentiment analysis. As highlighted in this study, Turkish exhibits rich morphology, agglutinative structure, and specific word order patterns, which can significantly impact sentiment analysis tasks. These linguistic nuances require specialized techniques to capture sentiment nuances accurately. For instance, the proposed task-specific transformations, such as sentence order swap and extending adjustment, directly address the intricacies of Turkish sentence structures. The ensemble approach introduced in this research not only augments the dataset but also leverages these linguistic insights to enhance sentiment analysis performance. Understanding the role of Turkish linguistic features in sentiment analysis is vital for building models that can effectively capture the sentiment expressed in Turkish texts. For instance, the task of identifying sentiment polarity in Turkish sentences often depends on context, word order, and even subtle grammatical variations. By combining task-specific and universal transformations, our approach aims to bridge the gap between these linguistic intricacies and model capabilities. Future research in sentiment analysis for Turkish should continue to explore and exploit the language's unique characteristics. The adaptability of the proposed ensemble augmentation approach to model uncertainty, presents a promising avenue. By tailoring augmentation strategies to account for the uncertainty arising from Turkish language intricacies, we can further improve the reliability of sentiment analysis models for this language. Moreover, our findings in Turkish sentiment analysis also have broader implications for low-resource languages, where linguistic peculiarities demand specialized approaches. In summary, this study underscores the importance of acknowledging Turkish language characteristics in sentiment analysis and lays the foundation for further research into tailored augmentation techniques that can enhance model performance and adapt to the intricacies of this language.

In the realm of sentiment analysis, it is imperative to address potential biases stemming from linguistic diversity, particularly in a language as rich and varied as Turkish. Turkish, spoken across diverse regions and communities, exhibits a myriad of dialects, each with its distinct lexicon, expressions, and nuances. These dialectal differences can profoundly influence sentiment and emotional expressions. Neglecting this diversity may inadvertently lead to biases in sentiment analysis models. To mitigate this, researchers need to adopt a more inclusive approach during data collection and augmentation. This entails incorporating representative samples from various Turkish dialects, acknowledging that expressions of sentiment can significantly differ between regions. Furthermore, understanding the impact of specific use cases on sentiment analysis is paramount. Sentiment can vary significantly based on context, and the choice of use case or domain can introduce biases. For example, sentiment analysis in e-commerce may involve unique product-related terminology, whereas sentiment analysis in social media may entail distinct linguistic features driven by internet culture and slang. Failure to consider these variations could result in models that excel in one domain but falter in others. To address these potential biases comprehensively, researchers should adopt a holistic approach. This begins with meticulous dataset curation, ensuring that dialectal diversity is adequately represented and that domain-specific nuances are considered. Augmentation techniques should be tailored to respect and incorporate these variations, thus producing augmented data that retains the linguistic richness of Turkish discourse. Additionally, analyzing the performance of sentiment analysis models across different dialects and domains can shed light on potential biases and inform model improvements. Embracing the linguistic diversity of Turkish, encompassing dialects, and acknowledging domain-specific influences are essential steps in developing robust sentiment analysis models. By doing so, researchers can navigate potential biases and create models that not only excel in various contexts but also respect and represent the multifaceted nature of the Turkish language.

The application of synthetic text generation techniques raises a host of ethical concerns that warrant careful consideration. While these methods offer promising avenues for enhancing the performance of natural language processing models, they also carry potential risks and responsibilities that researchers and practitioners must address.

Bias and Fairness: One primary concern is the inadvertent introduction or amplification of bias within augmented text data. If the original data used for augmentation contains biases, these biases can propagate into the synthetic data, further exacerbating disparities and unfairness. This has significant implications, particularly in sentiment analysis, where biased or skewed data can lead to inaccurate sentiment predictions. Researchers must be vigilant in identifying and mitigating biases to ensure that synthetic data does not perpetuate or amplify societal prejudices.

Privacy and Confidentiality: Synthetic text generation may inadvertently expose sensitive or private information. Even when trained on publicly available data, generated text could potentially reveal personal details about individuals or organizations. Protecting privacy is paramount, and researchers must implement measures to anonymize or obfuscate any sensitive content in the synthetic data.

Content Quality and Misinformation: Generating text that mimics human language can inadvertently produce content that is misleading, inaccurate, or even harmful. Ensuring the quality and accuracy of synthetic text is crucial, as it may be used for various downstream applications, including sentiment analysis. Rigorous validation and review processes are necessary to identify and rectify any inaccuracies or misleading information in the augmented data.

Transparency and Accountability: Responsible research and practice in synthetic text generation require transparency and accountability. Researchers should document their

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

augmentation methods, data sources, and processes comprehensively. Sharing this information not only aids in reproducibility but also allows for scrutiny and validation of the generated data.

Guidelines and Best Practices: As synthetic text generation evolves, the research community should develop and promote ethical guidelines and best practices. These guidelines can serve as a roadmap for researchers and practitioners, offering insights into mitigating bias, ensuring privacy, and maintaining the quality of synthetic text data.

Ongoing Ethical Discussions: Given the potential societal impact of synthetic text in applications like sentiment analysis, ongoing ethical discussions are vital. These discussions should involve interdisciplinary collaboration, including experts in ethics, privacy, linguistics, and computer science. Together, we can establish a framework for responsible synthetic text generation that prioritizes fairness, privacy, and the ethical use of artificial text data.

While synthetic text generation holds significant promise for advancing natural language processing, it must be approached with a strong ethical foundation. Acknowledging these ethical considerations is the first step in ensuring that the benefits of augmented data are realized without compromising fairness, privacy, or the quality of the information generated. Researchers and practitioners alike must be proactive in addressing these concerns, and the research community should continue to engage in ethical discussions to guide the responsible development and deployment of synthetic text techniques.

A failure analysis involves identifying patterns or observations where the performance of the language models decreases or remains relatively low despite different transformation functions. Here are some observations and discussions based on the empirical results:

Consistency Across Models: The proposed scheme consistently achieves the highest performance scores across most language models. This suggests that the proposed scheme is robust and effective in improving model performance.

General Lack of Improvement: Across all transformation functions, there's a noticeable trend that performance improvements are relatively modest. For example, even with sophisticated augmentation techniques, the scores remain in the range of 0.86 to 0.93, indicating that improving model performance is a challenging task.

Limited Impact of Some Techniques: Some augmentation techniques, such as "Swap Named Entity (SNE)" and "No Augmentation," show limited impact on model performance. These techniques consistently yield lower scores compared to others, indicating that they may not be very effective for the given task.

Sensitivity to Specific Models: The effectiveness of transformation functions can vary depending on the underlying language model. For instance, "BERT-Large-ITPT" consistently performs well with most augmentation techniques, while "ConvBERTurk" lags behind in several cases. This suggests that the choice of the base model can significantly affect augmentation outcomes.

Plateauing of Performance: After a certain point, further augmentation doesn't seem to significantly improve performance. For instance, while the proposed scheme generally outperforms other techniques, the scores tend to plateau around 0.92 to 0.93.

Variability in Impact: Some transformation functions, like "Spelling Error (SE)" and "SOS + DD," show varying impacts across different models. This suggests that certain techniques might be more sensitive to the specifics of the language model or the nature of the data.

High Robustness: Despite variations in techniques, all models consistently score relatively high, indicating that the base language models are already proficient in the task. This could mean that further improvements would require more advanced techniques or potentially different evaluation metrics.

Consideration of Resource Constraints: It's essential to consider resource constraints when choosing augmentation techniques. Some techniques might be computationally expensive or impractical for certain applications, so a trade-off between performance gains and resource requirements should be considered. While the proposed scheme consistently performs well, achieving substantial improvements in model performance beyond a certain point remains a challenging task. The choice of augmentation technique should be made with careful consideration of the specific language model and task requirements. Additionally, exploring more advanced techniques or other aspects of model training and evaluation may be necessary to achieve further improvements in performance.

The empirical results obtained through the extensive evaluation of eight state-of-the-art NLP models, namely XL-Net, ULM-Fit, BERT-Large-ITPT, SSL, BERTurk, DistilBERTurk, ELECTRA, and ConvBERTurk, provide valuable insights into their performance under various data augmentation techniques. This discussion aims to highlight key findings, patterns, and areas of concern observed during the evaluation. Our analysis reveals that all models exhibit varying degrees of sensitivity to the choice of data augmentation techniques. Notably, techniques such as "Spelling Error (SE)," "Sentence Order Swap (SOS)," and "Double Denial (DD)" consistently yield improved performance across models, suggesting their efficacy in enhancing model robustness. However, certain models, including ULM-Fit and BERTurk, display limited sensitivity to these techniques, indicating potential shortcomings in adapting to certain types of data transformations. One noteworthy observation is the performance discrepancy between models when transitioning from augmented data to the original task. While some models demonstrate robust generalization, others struggle to maintain their augmented performance levels on unaltered data. This phenomenon underscores the importance of evaluating a model's capacity to generalize effectively, particularly in real-world scenarios where the distribution of data may vary. Comparing the performance of models pretrained on diverse corpora, we find variations in their effectiveness across augmentation techniques. For instance, BERT-Large-ITPT consistently exhibits competitive performance, highlighting the potential benefits of leveraging pretraining on a substantial

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

amount of data. Conversely, ULM-Fit, while effective in certain cases, faces challenges in maintaining high performance across augmentation techniques. This variation emphasizes the significance of pretrained model selection tailored to the specific NLP task. Our analysis unveils variations in the impact of different augmentation techniques on model performance. Techniques such as "BERT Augment" and "Synonym Replacement" tend to have a positive influence across most models, while "Keyboard Error" exhibits mixed effects. Understanding how these techniques affect model behavior can guide practitioners in selecting appropriate data augmentation strategies. Considering resource constraints in practical applications, we observe that models like DistilBERTurk and ConvBERTurk, which require fewer computational resources, perform competitively with their resource-intensive counterparts. This finding suggests the feasibility of resource-efficient model choices without significant sacrifices in performance. Certain models exhibit a heightened sensitivity to specific types of data noise introduced by augmentation techniques, leading to a decrease in overall performance. Identifying these sensitivities is crucial for addressing model limitations and devising mitigation strategies. While our evaluation provides insights into model behavior on benchmark datasets, the real-world applicability of these findings must be considered. Models that perform well in controlled settings may not necessarily excel when applied to diverse, real-world data. Future research should prioritize assessing model generalization to ensure practical utility. This comprehensive empirical analysis offers a nuanced understanding of the strengths and weaknesses of state-of-the-art NLP models under various data augmentation techniques. The observed variations underscore the need for careful model selection, adaptation, and augmentation strategy design, with a focus on enhancing generalization and robustness in real-world NLP applications. These findings provide valuable guidance to researchers and practitioners seeking to optimize model performance in diverse NLP tasks.

While our research has yielded promising results and contributions to the field of sentiment analysis, it is essential to acknowledge several limitations that should be considered:

1. Dataset Size: The size of the datasets used, while substantial, may still limit the generalizability of our findings to a broader range of Turkish sentiment analysis tasks. Larger datasets would provide a more comprehensive evaluation.
2. Language Complexity: Turkish is a linguistically complex language with various nuances and contextual variations. While our augmentation techniques aim to address these complexities, challenges related to the rich linguistic structure of Turkish remain.
3. Model Choices: While we have employed state-of-the-art models for sentiment analysis, there are various model architectures and pre-trained embeddings available. Alternative model choices could yield different results.
4. Ethical Considerations: While we have emphasized ethical data handling, it is important to recognize that data augmentation, if not performed responsibly, can introduce biases or privacy concerns. Further research is needed in ethical augmentation practices.
5. Resource Constraints: The computational resources required for training large-scale models can be substantial. Resource-constrained environments may face challenges in replicating our experiments.
6. Generalization Beyond Datasets: While we have conducted experiments on a diverse set of datasets, including low-resource languages, the generalizability of our augmentation techniques to all possible sentiment analysis scenarios may require further investigation.

VI. CONCLUSIONS

In conclusion, our research on text augmentation techniques for Turkish sentiment analysis has yielded valuable insights and contributions to the field of natural language processing. We have demonstrated the effectiveness of our ensemble method, which combines multiple transformation functions, in improving sentiment analysis accuracy across diverse datasets. Key Takeaways: Ensemble Approach: Our ensemble method, which incorporates a combination of transformation functions, has consistently shown superior performance in sentiment analysis tasks. It provides a robust solution for addressing the challenges of Turkish sentiment analysis. Tailored Applicability: Our research highlights the tailored applicability of augmentation techniques to Turkish sentiment analysis, considering the linguistic characteristics of the Turkish language. This tailored approach offers a novel perspective in the field. Ethical Data Augmentation: We have emphasized the importance of ethical data augmentation practices, including bias mitigation, transparency, and privacy considerations. Responsible AI practices are integral to the success of data augmentation techniques. Practical Implications: Our findings have practical implications for organizations and decision-makers involved in sentiment analysis tasks. Our techniques can enhance customer feedback analysis, social media sentiment monitoring, and product review assessment. Limitations and Future Directions: We have transparently acknowledged the limitations of our research, including dataset size and language complexity. These limitations open avenues for future research to address these challenges and further improve sentiment analysis in Turkish. In summary, our research contributes to the advancement of sentiment analysis techniques for Turkish, and the ensemble approach we propose holds promise for broader applications in natural language processing. We invite further exploration and experimentation in this domain to build upon our findings and continue to enhance sentiment analysis capabilities.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

REFERENCES

[1] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Text data augmentation for deep learning," Journal of Big Data, vol. 8, pp. 1–34, 2021.
[2] P. Liu, X. Wang, C. Xiang, and W. Meng, "A survey of text data augmen- tation," 2020 International Conference on Computer Communication and Network Security (CCNS), pp. 191–195, 2020.
[3] B. Li, Y. Hou, and W. Che, "Data augmentation approaches in natural language processing: A survey," AI Open, vol. 3, pp. 71–90, 2022.
[4] M. Bayer, M. A. Kaufhold, and C. Reuter, "A survey on data augmentation for text classification," ACM Computing Surveys, vol. 55, no. 7, pp. 1–39, 2022.
[5] A. Ragni, K. M. Knill, S. P. Rath, and M. J. Gales, "Data augmentation for low resource languages," INTERSPEECH 2014: 15th Annual Conference of the International Speech Communication Association, pp. 810–814, 2014.
[6] G. Eryig˘it, "ITU Turkish NLP web service," Proceedings of the Demon- strations at the 14th Conference of the European Chapter of the Associa- tion for Computational Linguistics, pp. 1–4, 2014.
[7] E. Yıldırım, F. S. Çetin, G. Eryig˘it, and T. Temel, "The impact of NLP on Turkish sentiment analysis. Türkiye Bilis¸im Vakfı Bilgisayar Bilimleri ve Mühendislig˘i Dergisi," arXiv preprint, vol. 7, pp. 43–51, 2015.
[8] Oflazer, K., Saraçlar, and M., Eds., Turkish natural language processing. Springer International Publishing, 2018.
[9] A. Onan, "Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks," Concurrency and Computation: Practice and Experience, vol. 33, no. 23, pp. 5909–5909, 2021.
[10] Y. Belinkov and Y. Bisk, "Synthetic and natural noise both break neural machine translation," arXiv preprint, 2017.
[11] S. Y. Feng, V. Gangal, D. Kang, T. Mitamura, and E. Hovy, "Genaug: Data augmentation for finetuning text generators," arXiv preprint, 2020.
[12] C. Coulombe, "Text data augmentation made simple by leveraging NLP cloud APIs," arXiv preprint, 2018.
[13] J. Wei and K. Zou, "EDA: Easy data augmentation techniques for boosting performance on text classification tasks," arXiv preprint, 2019.
[14] Y. Li, T. Cohn, and T. Baldwin, "Robust training under linguistic adver- sity," Proceedings of the 15th Conference of the European Chapter, vol. 2, pp. 21–27, 2017.
[15] G. Rizos, K. Hemker, and B. Schuller, "Augment to prevent: short-text data augmentation in deep learning for hate-speech classification," Proceedings of the 28th ACM international conference on information and knowledge management, pp. 991–1000, 2019.
[16] X. Sun and J. He, "A novel approach to generate a large scale of supervised data for short text sentiment analysis," Multimedia Tools and Applications, vol. 79, no. 9, pp. 5439–5459, 2020.
[17] O. Kolomiyets, S. Bethard, and M. F. Moens, "Model-portability exper- iments for textual temporal analysis," Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Lan- guage Technologies, 2011.
[18] A. Mosolova, V. Fomin, and I. Bondarenko, "Text Augmentation for Neural Networks. ," AIST (Supplement), pp. 104–109, 2018.
[19] W. Y. Wang and D. Yang, "That's so annoying!!!: A lexical and frame- semantic embedding based data augmentation approach to automatic cate- gorization of annoying behaviors using# petpeeve tweets," Proceedings of the 2015 conference on empirical methods in natural language processing, pp. 2557–2563, 2015.
[20] S. Y. Feng, V. Gangal, D. Kang, T. Mitamura, and E. Hovy, "Genaug: Data augmentation for finetuning text generators," arXiv preprint, 2020.
[21] G. G. S¸ ahin and M. Steedman, "Data augmentation via dependency tree morphing for low-resource languages," arXiv preprint, 2019.
[22] G. G. Sahin, "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, 2022.
[23] S. Rahmani, S. Hosseini, R. Zall, M. R. Kangavari, S. Kamran, and W. Hua, "Transfer-based adaptive tree for multimodal sentiment analysis based on user latent aspects. ," Knowledge-Based Systems, vol. 261, pp. 110 219–110 219, 2023.
[24] S. Najafipour, S. Hosseini, W. Hua, M. R. Kangavari, and X. Zhou, "Soul- Mate: Short-text author linking through Multi-aspect temporal-textual embedding," IEEE Transactions on Knowledge and Data Engineering, vol. 34, no. 1, pp. 448–461, 2020.
[25] M. Saaki, S. Hosseini, S. Rahmani, M. R. Kangavari, W. Hua, and
X. Zhou, "Value-wise ConvNet for Transformer models: An Infinite Time- aware Recommender System," IEEE Transactions on Knowledge and Data Engineering, 2022.
[26] S. Kamran, R. Zall, S. Hosseini, M. Kangavari, S. Rahmani, and W. Hua, "EmoDNN: understanding emotions from short texts through a deep neural network ensemble," Neural Computing and Applications, pp. 1–18, 2023.
[27] X. Wang, Q. Liu, T. Gui, Q. Zhang, Y. Zou, X. Zhou, . . Huang, and X. J, "Textflint: Unified multilingual robustness evaluation toolkit for natural language processing," Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations, pp. 347–355, 2021.
[28] Ö. Bakay, Ö. Ergelen, E. Sarmıs¸, S. Yıldırım, B. N. Arıcan, A. Kocabal- cıog˘lu, . . Yıldız, and O. T, "Turkish wordnet kenet," Proceedings of the 11th global wordnet conference, pp. 166–174, 2021.
[29] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understand- ing," Advances in neural information processing systems, pp. 32–32, 2019.
[30] W. Lan, Y. Chen, W. Xu, and A. Ritter, "An empirical study of pre-trained transformers for Arabic information extraction," arXiv preprint, 2020.
[31] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," arXiv preprint, 2018.
[32] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint, 2018.
[33] D. S. Sachan, M. Zaheer, and R. Salakhutdinov, "Revisiting lstm networks for semi-supervised text classification via mixed objective function," Pro- ceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 6940–6948, 2019.
[34] S. Schweter, "Berturk-bert models for turkish," arXiv preprint, pp. 3 770 924–3 770 924, 2020.
[35] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," arXiv preprint, 2019.
[36] K. Clark, M. T. Luong, Q. V. Le, C. D. Manning, and arXiv preprint, "Elec- tra: Pre-training text encoders as discriminators rather than generators," Electra: Pre-training text encoders as discriminators rather than generators, 2020.
[37] Z. H. Jiang, W. Yu, D. Zhou, Y. Chen, J. Feng, and S. Yan, "Convbert: Im- proving bert with span-based dynamic convolution," Advances in Neural Information Processing Systems, vol. 33, pp. 12 837–12 848, 2020.
[38] M. Aydog˘an and V. Kocaman, "TRSAv1: a new benchmark dataset for classifying user reviews on Turkish e-commerce websites," Journal of Information Science, pp. 01 655 515 221 074 328–01 655 515 221 074 328, 2022.
[39] A. Onan, "Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks," Concurrency and Computation: Practice and Experience, vol. 33, no. 23, pp. 5909–5909, 2021.
[40] A. Kamal and M. Abulaish, "Cat-BiGRU: Convolution and attention with bi-directional gated recurrent unit for self-deprecating sarcasm detection," Cognitive Computation, vol. 14, no. 6, pp. 1657–1675, 2022.
[41] A. Onan, "Bidirectional convolutional recurrent neural network architec- ture with group-wise enhancement mechanism for text sentiment clas- sification," Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 5, pp. 2098–2117, 2022.
[42] S. Schweter, "Berturk-bert models for Turkish," arXiv preprint, pp. 3 770 924–3 770 924, 2020.
[43] G. Rizos, K. Hemker, and B. Schuller, "Augment to prevent: Short- text data augmentation in deep learning for hate-speech

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3349971

classification," Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 991–1000, 2019.

[44] L. M. R. Rizky and S. Suyanto, "Improving stance-based fake news detection using BERT model with synonym replacement and random swap data augmentation technique," 2021 IEEE 7th Information Technology International Seminar (ITIS), pp. 1–6, 2021.

[45] K. He, R. Mao, T. Gong, C. Li, and E. Cambria, "Meta-based self-training and re-weighting for aspect-based sentiment analysis," IEEE Transactions on Affective Computing, 2022.

[46] A. Y. Uluslu, "Automatic lexical simplification for Turkish," arXiv preprint, 2022.

[47] T. Hong, D. Lee, J. Han, S. J. Choi, and J. Seo, "CECADA: Cause- Effect Conjunctive Adverb-based Data Augmentation Method in Low- Resource Knowledge-Grounded Dialogue," Information Sciences, vol. 574, pp. 574–592, 2023.

[48] C. Toraman, I. E. Kucukkaya, O. Ozcelik, and U. Sahin, "Tweets Under the Rubble: Detection of Messages Calling for Help in Earthquake Disaster," arXiv preprint, 2023.

[49] A. Onan, "GTR-GA: Harnessing the Power of Graph-Based Neural Networks and Genetic Algorithms for Text Augmentation". Expert Systems with Applications, 120908..

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4