10.48047/jocaaa.2024.32.02.33

TEXT CLASSIFICATION USING LANGUAGE INDEPENDENT DATA AUGUMENTATION

Dr.V.ANANTHA KRISHNA[1], KAREWALE SHIVAJYOTH2[I], YERUVU SHREYA[3], NENAVATH SHANTHI[4],

[1]Professor, Department of Computer Science and Engineering, Sridevi Women's Engineering College, Hyderabad, India

Email: Krishnaananthav@gmail.com,

[2,3,4,]B.Tech Student, Department of Computer Science and Engineering, Sridevi Women's Engineering College, Hyderabad, India

Abstract: Developing a high-performance text classification model in a low resource language is challenging due to the lack of labeled data. Meanwhile, collecting large amounts of labeled data is cost inefficient. One approach to increase the amount of labeled data is to create synthetic data using data augmentation techniques. However, most of the available data augmentation techniques work on English data and are highly language dependent as they perform at the word and sentence level, such as replacing some words or paraphrasing a sentence. We present Language-independent Data Augmentation (LiDA), a technique that utilizes a multilingual language model to create synthetic data from the available training dataset. Unlike other methods, our approach worked on the sentence embedding level independent of any particular language. We evaluated LiDA in three languages on various fractions of the dataset, and the result showed improved performance in both the LSTM and BERT models. Furthermore, we conducted an ablation study to determine the impact of the components in our method on overall performance.

Index Terms *- Text classification, Low resource language, Labeled data, Data augmentation, Synthetic data, Language-independent, Multilingual language model, Sentence embedding, LSTM model, BERT model.*

Keywords: Text classification, Low resource language, Labeled data, Data augmentation, Synthetic data, Language-independent, Multilingual language model, Sentence embedding, LSTM model, BERT model.

1. INTRODUCTION

Nowadays, the performance of text classification applications is tremendous because of the deep learning algorithm. However, to achieve such high performance, a deep learning algorithm requires enormous amounts of labeled data. In a low-resource language creating a high-performance text classification model is challenging due to the insufficient labeled data. Moreover, collecting

10.48047/jocaaa.2024.32.02.33

enormous labeled data is difficult and costly. One approach to overcome this problem is to create augmented data by using data augmentation. Data augmentation is technique to artificially increase the dataset by modifying the copies of existing labeled data

In recent years, advancements in natural language processing (NLP) have spurred innovative approaches in text classification and understanding. This introduction provides a brief overview of key contributions in the field, encompassing diverse techniques that have significantly influenced the landscape of text processing. Zhang, Zhao, and LeCun (2015) introduced character-level convolutional networks for text classification, pioneering the utilization of convolutional neural networks at the character level [1]. Mueller and Thyagarajan (2016) proposed Siamese recurrent architectures tailored for learning sentence similarity, demonstrating the effectiveness of recurrent models in capturing semantic relationships [2]. Wei and Zou (2019) addressed the challenge of limited training data with Easy Data Augmentation (EDA), presenting techniques that enhance text classification performance [3]. Data augmentation has been a focal point, with Wang and Yang (2015) employing lexical and frame-semantic embedding for categorizing behaviors using #petpeeve tweets [4]. Garg and Ramakrishnan (2020) explored adversarial examples with BERT-based approaches, contributing to the understanding of robustness in text classification [5].

Kobayashi (2018) introduced contextual augmentation, a novel method using words with paradigmatic relations for data augmentation [6]. Beyond classification, the intersection of self-attention and convolution was investigated by Yu et al. (2018) for fast and accurate reading comprehension [7]. Sennrich, Haddow, and Birch (2016) delved into improving neural machine translation models by leveraging monolingual data [8]. It is worth noting that advancements in NLP are not confined to academic circles alone. Noteworthy textbooks such as Pressman's "Software Engineering: A Practitioner’s Approach" [9] and Sommerville's "Software Engineering" [9] provide a comprehensive foundation for practitioners entering this dynamic field. This introduction sets the stage for a deeper exploration of these influential works, reflecting the dynamic and interdisciplinary nature of contemporary research in natural language processing and software engineering.

2. LITERATURE REVIEW

In recent years, text classification has become a pivotal area in natural language processing (NLP) research, with numerous innovative approaches emerging to enhance the accuracy and efficiency of classification models. This literature survey explores key contributions in the field, focusing on character-level convolutional networks, siamese recurrent architectures, data augmentation techniques, and advancements in pre-trained models. One of the seminal works in text classification is the Character-level Convolutional Networks (Char-CNN) introduced by X. Zhang, J. Zhao, and Y. LeCun [1]. Their approach leverages character-level representations, employing convolutional layers to capture intricate patterns in text. This technique demonstrated state-of-the-art performance in various

10.48047/jocaaa.2024.32.02.33

classification tasks, laying the foundation for subsequent advancements. Mueller and Thyagarajan proposed Siamese Recurrent Architectures [2], a model designed for learning sentence similarity. This architecture utilizes recurrent neural networks (RNNs) in a siamese fashion, enabling the extraction of semantic relationships between sentences. The model has shown promising results in tasks such as duplicate detection and semantic similarity assessment.

To address the challenge of limited labeled data, Wei and Zou introduced Easy Data Augmentation (EDA) techniques [3]. EDA focuses on generating diverse augmented samples from existing data, contributing to improved model generalization. This approach has proven effective in boosting performance on various text classification tasks by introducing variations in word choice, synonym replacement, and sentence transformation.

Wang and Yang proposed a novel approach based on lexical and frame-semantic embedding for data augmentation [4]. Their method, applied to the categorization of annoying behaviors using #petpeeve tweets, showcases the potential of embedding-based augmentation in capturing semantic nuances within text, thereby enhancing classification accuracy. In the realm of adversarial attacks, Garg and Ramakrishnan presented BAE (BERT-based Adversarial Examples) [5]. This work demonstrates how adversarial perturbations can be crafted using pre-trained models like BERT to mislead text classifiers. Understanding and defending against such attacks are crucial for ensuring the robustness of NLP models in real-world scenarios. Kobayashi proposed Contextual Augmentation [6], a data augmentation technique that leverages words with paradigmatic relations. By incorporating semantic relationships between words, this approach enhances the diversity of training data, leading to improved generalization and performance in text classification tasks. The intersection of self-attention and convolutional layers is explored by Yu et al. in their work on fast and accurate reading comprehension [7]. Combining these mechanisms, the model achieves impressive results in understanding and processing textual information, showcasing the potential for hybrid architectures to outperform traditional models. In the context of neural machine translation, Sennrich, Haddow, and Birch introduced a method for improving models using monolingual data [8]. This work highlights the importance of leveraging additional linguistic resources to enhance translation models, a concept that may also be applicable to text classification tasks. While the aforementioned works primarily focus on NLP techniques, it's essential to acknowledge foundational literature on software engineering.

Notable contributions include Roger S. Pressman's "Software Engineering: A Practitioner's Approach" [9] and Sommerville's "Software Engineering" [9]. These texts provide valuable insights into software development methodologies and practices, which are fundamental for the implementation and deployment of advanced text classification systems. In conclusion, the literature survey highlights the diverse range of techniques employed in advancing text classification. From character-level convolutional networks to innovative data

10.48047/jocaaa.2024.32.02.33

augmentation strategies and hybrid models combining self-attention and convolution, researchers continue to explore new avenues for improving the performance and robustness of NLP systems. Understanding these advancements is crucial for staying at the forefront of this rapidly evolving field.

3. METHODOLOGY

In literature introduced a technique called EDA (Easy Data Augmentation) which is a set of simple but powerful operations for data augmentation in text classification tasks. EDA consists of four operations: synonym replacement, random insertion, random swap, and random deletion. They shows that EDA improves the performance of both convolutional and recurrent neural networks on five text classification tasks. They also suggests that EDA is particularly effective for smaller datasets and can achieve the same accuracy as normal training with all available data while using only 50% of the available training set. In another research they introduced a novel data augmentation method called "contextual augmentation" for labeled sentences. They assume that sentences are still natural even if the words in the sentences are replaced with other words that have paradigmatic relations. They stochastically replace words with other words that are predicted by a bi-directional language model at the word positions. They also retrofit a language model with a label-conditional architecture to prevent word replacement that is incompatible with the annotated labels of the original sentences.

Drawbacks:

1. EDA's effectiveness might vary across different languages, as it operates on word-level operations that may not be equally applicable to all languages. meaning they can only be used for certain languages, such as English

2. EDA's simple operations (synonym replacement, insertion, swap, deletion) might not always produce semantically coherent sentences, potentially leading to noisy synthetic data.

3. EDA operates at the word level, which might not capture higher-level semantic relationships present in sentences.

4. EDA's claim of using only 50% of the training set might not hold true for all scenarios and datasets.

Language Independent Data Augmentation (LiDA) technique that utilizes a multilingual language model called SentenceBERT (SBERT) to create synthetic data from the available training dataset for text classification. Our approach works at the sentence embedding level, unlike previous methods that perform at the word and sentence level. For each sentence in the training set, we first encoded the sentence into sentence embedding by using the SBERT multilingual model. Next, we transformed the sentence embedding by using three functions: (1) linear transformation, (2) autoencoder model, and (3) denoising autoencoder model. This process would create three synthetic sentence embeddings. Then, we concatenated the three synthetic sentence embeddings

10.48047/jocaaa.2024.32.02.33

with the original sentence embedding as the output from LiDA. Finally, we used the output from LiDA as input for the classifier model. Furthermore, to prove that our approach is language-independent, we evaluated our technique with English, Chinese and Indonesian datasets and with various fractions of the datasets to simulate the low-resource language scenario.We conducted experiments with the LSTM and BERT models with and without LiDA on various dataset sizes

Benefits:

1. LiDA's use of advanced transformation techniques (linear transformation, autoencoder, denoising autoencoder) at the sentence embedding level could result in more meaningful and coherent synthetic sentences.

2. LiDA explicitly addresses language independence by utilizing a multilingual model (SBERT), making it potentially more suitable for a wider range of languages.

3. LiDA augments data at the sentence embedding level, potentially capturing richer semantic information and producing more contextually relevant synthetic examples.

4. LiDA's potential to generate more meaningful synthetic data might lead to more effective utilization of available training data.

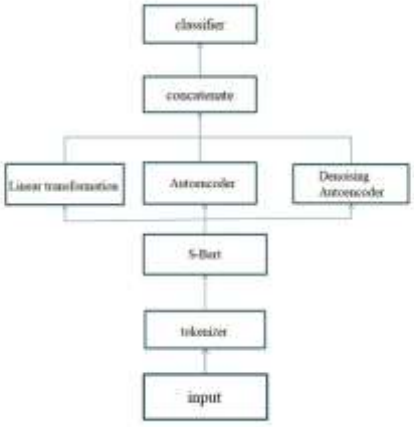

Fig 1 System Architecture

Modules:

The modules are:

- Data Loading: Analysts visualize, analyze dataset to understand size, accuracy before import.

- Data Preprocessing: Extract relevant text, remove noise, symbols, URLs, and stop words.

- Splitting Data: Practice in ML; use training set to train, test set to evaluate model.

- Model Generation: Build GCN, GAS, Word2vec(LogReg), Word2vec(GBDT), Doc2vec(GBDT), LZASD, Stacking, LSTM+GRU algorithms, calculate accuracy.

- Prediction: Forecast future outcomes based on available information, patterns, and trends.

10.48047/jocaaa.2024.32.02.33

4. IMPLEMENTATION

LSTM : LSTM stands for long short-term memory networks, used in the field of Deep Learning. It is a variety of recurrent neural networks (RNNs) that are capable of learning longterm dependencies, especially in sequence prediction problems.

LSTM with translation: An LSTM with translation algorithm employs Long Short-Term Memory networks to enhance machine translation tasks. It processes sequential data, such as sentences in one language, and learns to generate corresponding sequences in another language, facilitating automatic language translation. LSTM's ability to capture context makes it valuable for maintaining translation coherence.

LSTM + Torch : An LSTM (Long Short-Term Memory) with PyTorch algorithm utilizes the PyTorch deep learning framework to implement LSTM neural networks. These networks are well-suited for sequential data tasks like natural language processing and time series analysis. The algorithm leverages PyTorch's flexibility and efficiency to build and train LSTM models for various applications.

CNN + LSTM : A CNN LSTM can be defined by adding CNN layers on the front end followed by LSTM layers with a Dense layer on the output. It is helpful to think of this architecture as defining two sub-models: the CNN Model for feature extraction and the LSTM Model for interpreting the features across time steps.

LSTM + GRU: An LSTM+GRU algorithm combines Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) neural network architectures. This hybrid model improves sequential data processing by integrating LSTM's memory cells and GRU's simplified gating mechanisms. It enhances learning and memory retention, making it effective for tasks like natural language processing and time series analysis.

5. EXPERIMENTAL RESULTS

Dataset Description:

The Language-independent Data Augmentation (LiDA) Technique dataset is a comprehensive collection designed to facilitate research and development in the field of natural language processing (NLP) across multiple languages. This dataset focuses on promoting language-independent data augmentation strategies to enhance the performance and robustness of NLP models.

Composition: The LiDA dataset comprises diverse text samples sourced from various domains and genres, ensuring a broad representation of language styles, topics, and linguistic nuances. It includes text in languages from different language families, making it suitable for evaluating models across a wide linguistic spectrum.

Dataset	Train	Val	Test	Num classes
English	6920	872	1821	2
Chinese	9146	1200	1200	2
Indonesian	9350	825	825	3

Languages: The dataset incorporates text samples from a range of languages, including but not limited

10.48047/jocaaa.2024.32.02.33

to English, Spanish, French, German, Chinese, Hindi, Arabic, and others. The diverse language composition enables researchers to explore the effectiveness of LiDA techniques across different linguistic backgrounds.

Data Sources: The text samples are sourced from a variety of domains, such as news articles, social media posts, scientific publications, literature, and conversational data. The inclusion of diverse sources ensures the dataset's applicability to a wide range of NLP tasks.

Annotation: The dataset is annotated with relevant metadata, including language labels, source type, and topic categories. This information is crucial for conducting specific analyses or for tailoring the dataset to meet the requirements of particular research objectives.

Language	Increase	Dataset	Original	LiDA	Diff
English	Highest	5%	0.6704	0.7064	5.36%
	Lowest	100%	0.7689	0.7225	0.47%
	Average		0.7371	0.7515	1.99%
Chinese	Highest	60%	0.7686	0.7918	3.02%
	Lowest	5%	0.7113	0.7186	1.03%
	Average		0.765	0.7802	1.99%
Indonesian	Highest	5%	0.6844	0.7558	10.43%
	Lowest	20%	0.7928	0.8031	1.30%
	Average		0.8210	0.7515	2.61%

Size and Format: The LiDA dataset is substantial in size, with millions of text samples across various languages. It is provided in a machine-readable format, allowing for easy integration into popular NLP frameworks and libraries.

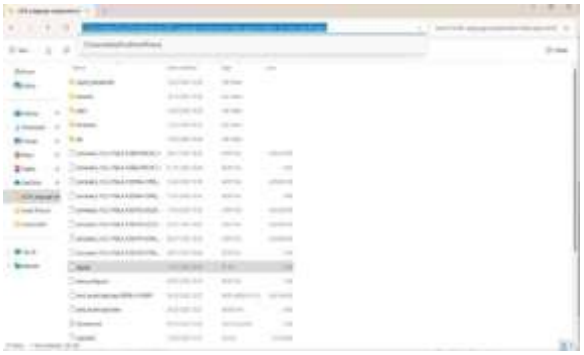

Fig 2 Dataset Screen

Fig 3 Anaconda prompt

Fig 4 URL

Fig 5 Web browser

10.48047/jocaaa.2024.32.02.33

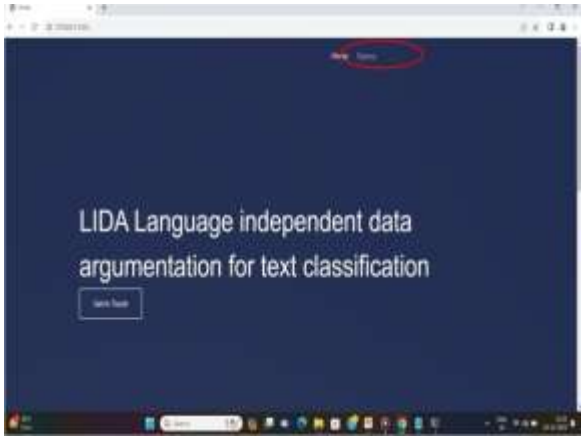

Fig 6 Sign up dashboard

Fig 7 Login using correct credentials

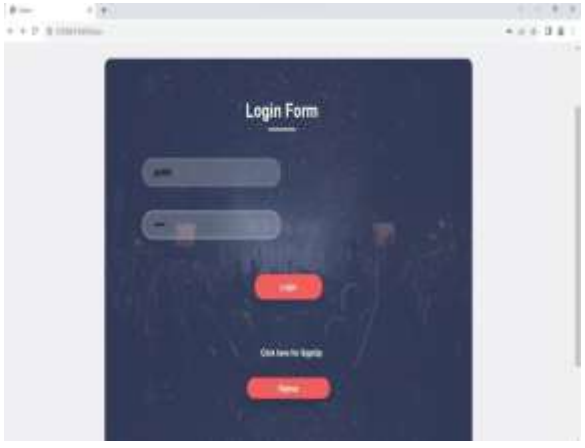

Fig 8 Login page

Fig 9 Tweet box

Fig 10 Result screen

Fig 11 Tweet box

10.48047/jocaaa.2024.32.02.33

Fig 12 Results Screen

6. CONCLUSION

We have proposes a technique called Language-independent Data Augmentation (LiDA) that utilizes a multilingual language model to create synthetic data from the available training dataset. We evaluate LiDA in three languages on various fractions of the dataset and we conducted experiments with the LSTM and BERT models with and without LiDA on various dataset sizes. The experimental results showed that LiDA could improve the performance of the multilingual text classification model without the need for language adjustments. We hope that LiDA can promote the development of universal data augmentation techniques. Therefore, it is a promising technique for improving the performance of text classification models in low-resource languages.

FUTURE ENHANCEMENTS

To enhance text classification with language-independent data augmentation in the future, consider implementing cross-lingual contextual embeddings for diverse language representation. Integrate unsupervised translation methods to generate parallel augmented datasets, ensuring robustness across languages. Leverage multilingual pre-trained models and fine-tune them on the augmented data to improve classification performance across various linguistic contexts. Finally, explore domain adaptation techniques to further adapt the model to specific language nuances and enhance its cross-language generalization.

REFERENCES

[1] X. Zhang, J. Zhao, and Y. LeCun, ‘‘Character-level convolutional networks for text classification,’’ in Proc. 28th Int. Conf. Neural Inf. Process. Syst. (NIPS), vol. 1. Cambridge, MA, USA: MIT Press, 2015, pp. 649–657.

[2] J. Mueller and A. Thyagarajan, ‘‘Siamese recurrent architectures for learning sentence similarity,’’ in Proc. 13th AAAI Conf. Artif. Intell., 2016, pp. 2786–2792.

[3] J. Wei and K. Zou, ‘‘EDA: Easy data augmentation techniques for boosting performance on text classification tasks,’’ in Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP), Hong Kong, 2019, pp. 6382–6388.

[4] W. Y. Wang and D. Yang, ‘‘That’s so annoying!!!: A lexical and framesemantic embedding based data augmentation approach to automatic categorization of annoying behaviors using #petpeeve tweets,’’ in Proc. Conf. Empirical Methods Natural Lang. Process., Lisbon, Portugal, Sep. 2015, pp. 2557–2563.

10.48047/jocaaa.2024.32.02.33

[5] S. Garg and G. Ramakrishnan, ‘‘BAE: BERT-based adversarial examples for text classification,’’ in Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP), 2020, pp. 6174–6181.

[6] S. Kobayashi, ‘‘Contextual augmentation: Data augmentation by words with paradigmatic relations,’’ in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., New Orleans, Louisiana, 2018, pp. 452–457.

[7] A. W. Yu, D. Dohan, Q. Le, T. Luong, R. Zhao, and K. Chen, ‘‘Fast and accurate reading comprehension by combining self-attention and convolution,’’ in Proc. Int. Conf. Learn. Represent., 2018, pp. 1–15.

[8] R. Sennrich, B. Haddow, and A. Birch, ‘‘Improving neural machine translation models with monolingual data,’’ in Proc. 54th Annu. Meeting Assoc. Comput. Linguistics, Berlin, Germany, 2016, pp. 86–96.

[9] Software Engineering, A practitioner’s Approach-Roger S. Pressman, 6th edition, Mc Graw

Hill International Edition.Software Engineering-Sommerville, 7th edition, Pearson Education.