2024 4th International Conference on Advancement in Electronics & Communication Engineering (AECE)

Ensemble Sentiment Analysis Using Bi-LSTM and CNN

Puneet Singh Lamba
Department of Computer Science, Sri Guru Tegh Bahadur Khalsa College, University of Delhi, Delhi 110007, India
singhs.puneet@gmail.com

Achin jain*
Bharati Vidyapeeth's College of Engineering, New Delhi, India
achin.mails@gmail.com

Harsh Taneja
Department of Computer Science and Engineering, Graphic Era (Deemed to be) University, Dehradun, India
harshtaneja.cse@gmail.com

Prakhar Priyadarshi
Bharati Vidyapeeth's College of Engineering,
New Delhi, India
prakharpriya@gmail.com

Arvind Panwar
School of computing science engineering, Galgotias University, India
Arvind.nice3@gmail.com

Arun Kumar Dubey
Bharati Vidyapeeth's College of Engineering, New Delhi, India
arudubey@gmail.com

*corresponding : achin.mails@gmail.com

2024 4th International Conference on Advancement in Electronics & Communication Engineering (AECE) | 979-8-3503-6472-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/AECE62803.2024.10911211

Abstract: Sentiment analysis is an invaluable skill for categorizing or evaluating points of view of people from all over the world. Understanding how people feel about a certain issue, product, or service is helpful. Sentiment analysis has been a focus of several studies, with most methods based on natural language processing techniques. Nevertheless, determining sarcasm, negation, or ambiguity in language usage has always proven challenging. In this research, we present an ensemble approach that includes methods such as Keras Sequential CNN and Bi-Long Short-Term Memory (Bi-LSTM). Two different datasets, the YELP review dataset and the IMDB review dataset, were used to evaluate the model. The outcomes were good, and our model performs well in comparison to a few other methods used to a related problem. Even though Bi-LSTM and Keras Sequential CNN produced good results on their own, employing ensemble techniques improved them even more.

I. Introduction

Text plays a vital role in most Artificial Intelligence-based techniques as well as other real-world applications. The monitoring of opinions and emotions is very useful and Sentiment analysis is widely used nowadays for the same. This technique of sentiment analysis can be used both for personal as well as commercial purposes [1]. It helps the industry in being better by understanding user reviews.

There are many techniques that have been used in obtaining analysis of sentiments. Some of the popular ones include the Bag of Words technique [2], Deep learning-based techniques [3,10], lexicon-based techniques [4], etc. All these are considered as a text classification techniques. Hence, sentiment analysis is treated as a supervised learning algorithm. Almost all machine learning models including sentiment analysis models need input to be a fixed-size vector of numbers. So, to convert text sequences of words into a fixed-size vector, several deep learning-based techniques and NLP is widely used. Also, there have been advancements in NLP for the same.

The concept of multi-layered sentiment analysis[8] is also very important as sentiments sometimes change their values when multiple types of sentiments are present in a single sentence. For e.g., let us imagine a Yelp review (from a dataset we have used):

"The linguini was great, but the room was way too dark".

Here, there are two types of sentiments in this sentence as Linguini is termed to be great which is a positive sentiment but the room is said to be dark which is a negative sentiment. So, the overall sentiment can be said to be neutral.

In this paper, we have used a hybrid of two techniques namely Keras Sequential CNN and Bi-LSTM:

- Two highly effective labelled datasets namely IMDB Review and Yelp review datasets were used. The procedure used in these labelled data was found to be very efficient.
- Firstly, a model was built using Bi-LSTM for both datasets [5]. The results were satisfactory in the case of this technique. Then the model was saved.
- Then, a model was built using the Bert Tokenizer and Keras Module. This model showed better results than Bi-LSTM. This model was also saved.
- Finally, an ensemble of the above two techniques was performed. The results shown were quite satisfying.

II. Related Work

A. *Lexicon-Based Sentiment Classification:*

This is an unsupervised way of working. This is how lexicon-based methods are done. The earlier techniques mainly involved labelling of training data. This involved use of several techniques for firstly labelling of the data. This was avoided in the Lexicon-based technique, as the need for labelled data was not there. This technique is used to construct a resource for its own use. Mapping of words was done into a grouping (positive, negative, neutral, or private).

There exists a lexical supply for text classification, namely Senti-WordNet [12]. Another resource, Senti-HowNet is used for Chinese texts. The approach is to check each word in an input sentence starting from left to right. Also, for all words, a weight score is computed. "+1" determines a positive sentiment, "-1" denotes a negative sentiment and "0" stands for neutral sentiment. Feature learning is the ML technique that was mainly used up in this concept.

This approach can be split into dictionary-based and corpus-based approaches[7]. In a dictionary-based approach, a trivial collection of words for ideas is collected by hand as a seed. Then well-known dictionaries or thesaurus are used to expand the set of ideas by adding their own synonyms and antonyms. Recent names are added to the seed list. The process continues until words are not found in the dictionary. Finally, manual updates are performed to remove errors. One of these methods is proposed by Kim and Hovy. The main problem with this method is that it cannot find the names of the idea that pertains to the background and context.

B. *Deep-Learning Based Sentiment Classification:*

One of the most often used methods for presenting document vocabulary is word embedding[6]. It is capable of storing a word's context throughout a document, as well as semantic and grammatically resemblance associations with other words, and other characteristics. Deep learning techniques also use this technique of word embeddings and are used in a large collection of applications of text mining, quality assurance, recommendation, and classification. Several techniques of deep learning are further divided into RNN and CNN. Sentiment classification can make use of both these techniques. When CNN was used, the results were quite promising.

A binary-directional LSTM was then being used in sentiment analysis, which also gave good results. Keeping in mind the efficacy of deep learning practices, sentiment analysis is further being improvised. Also, the incorporation of lexicon-based clues was made to be used in the training of LSTM models[11]. The proposed methods have a new loss function and are connected in the form of polar words or certain different types of words.

III. DATASET USED

A. *Yelp Datasets:*

The Yelp dataset [14] is just a small fraction of companies, reviews, and user data that can be utilized for personal, educational, and scholarly purposes. Provided in JSON format, this resource can be utilized for educational purposes such as teaching students about databases, learning natural language processing (NLP), or as example production data for practicing mobile app development.

The content of this dataset contains of following 2 features:

- positive: if the document sentiment is positive then this value is 1.
- negative: if the document sentiment is negative then this value is 0.

C. *IMDB dataset:*

This dataset [13] for characteristic words preparing or text analysis. This is a dataset for twofold sentiment order having significantly additional information(data) than past standard datasets. We give a bunch of 25,000 profoundly polar film surveys for preparing and 25,000 for testing. Along these lines, anticipate the quantity of positive and negative surveys utilizing either order or profound learning calculations.

Dataset comprises of two attributes:

- positive: if the document sentiment is positive then this value is positive.
- negative: if the document sentiment is negative then this value is negative.

IV. METHODOLOGY

The proposed model means to recognize sentiments utilizing profound deep learning methods [15][16][17]. This segment portrays the different procedures applied to the datasets depicted in the above segment. The experimentation is performed utilizing an outfit model [18] to break down the assessment. The group models which incorporate Keras Sequential CNN and Bi-LSTM have been applied and analysed for the general precision of the procedures on the datasets examined in the segment above. Preceding to the real characterization of information, a choice of advances have been performed for information pre-preparing lastly for preparing the information. For experimental work the dataset is split into 80:20 train-test ratio. The flowchart given in figure 1 portrays the key steps used in our proposed methodology for sentiment Analysis.

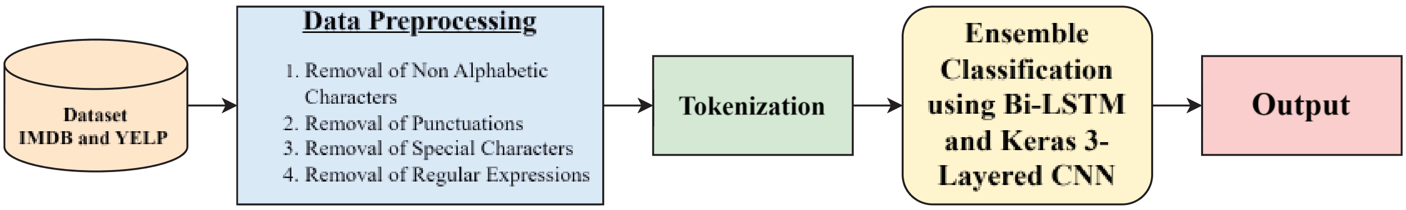

Fig. 1. Methodology Flow Chart

B. *Pre-processing*

Pre-processing includes data preprocessing [22][23][24] that alters the crude dataset into a conceivable configuration. It upgrades the information proficiency which influences the aftereffect of calculations. Moreover, data preprocessing is the most important part of a machine learning model. Different strides for data preprocessing include text cleaning, conversion of text, and removal of some particular type of numbers, characters, etc. Text Cleaning incorporates bringing in libraries, for example nltk, word_tokenize, stopwords, and Porter Stemmer for text cleaning. Conversion of text to lowercase is done to manage 26 letters in order as opposed to 52 letter sets to decrease the multifaceted nature of the model. Removal of numbers i.e., numbers is eliminated from the content as they are redundant in this examination. Removal of non-alphabetic characters means removal of different emoticons, images, emojis, and so on to get just letters in order for information investigation. The substance of each book is stacked and checked for the normal articulation to empower the cycle by eliminating all non-alphabetic characters. Tokenization is the way toward breaking the given content into more modest pieces called tokens. It is executed by the NLTK library's word_tokenize() work. Stop words are the most oftentimes utilized words in a language like 'a', 'is', 'the' and so forth. Since, these words don't have significant importance appended with themselves in this way, these words are commonly eliminated from the content. NLTK library is utilized to eliminate stop words. Stemming is the cycle of a decrease of words to their root or base structure. The doorman stemming calculation is utilized for the expulsion of normal morphological and in flexional endings from words.

V. Deep Learning Techniques

Deep learning is a subset of artificial intelligence that demonstrates superior results when applied to unstructured data. Profound learning enables computational models to continuously learn from the provided data. Computational models continuously adjust to do clustering tasks using provided text, audio, or images. Deep learning models can achieve state-of-the-art accuracy, sometimes even surpassing human performance.

A. Keras Sequential CNN 1D CNN Model

The Keras Python module facilitates the creation of deep learning models in a fast and effortless manner [22]. The consecutive API enables you to construct models incrementally, layer by layer, for a wide range of problems. It has limitations as it does not allow the creation of models with several layers or multiple inputs or outputs. The beneficial Application Programming Interface (API) in Keras serves as an alternative approach to constructing models, providing much enhanced flexibility, including the capacity to create more intricate models.

The Sequential model API, depicted in figure 2, is a technique for constructing deep learning models. It involves creating an instance of the Sequential class and subsequently adding model layers to it. The Sequential model API is incredible for growing deep learning models much of the time, however it likewise has a few impediments. For instance, it isn't clear to characterize models that may have numerous distinctive info sources, produce different yield objections, or models that re-use layers. The sequential API enables you to construct models in a step-by-step manner for various problems. The software has limitations that prevent users from creating models with multiple layers or incorporating diverse data sources or outputs.

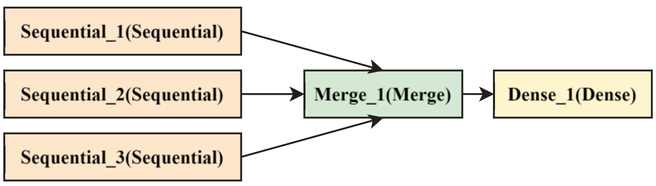

Fig. 2. Keras Sequential CNN model applied on text data

B. Bidirectional LSTM

A Bidirectional LSTM as shown in figure 3, also known as Bi-LSTM, is a sequential training model consisting of two LSTMs. One LSTM processes the input in a forward direction, while the other processes it in a backward direction. Bi-LSTMs effectively increase the amount of information available to the system, enhancing the context available to the algorithm (e.g. comprehending the words that immediately precede and follow a word in a sentence).

RNN structures like LSTM and Bi-LSTM are utilized in events where the learning issue is consecutive, for example, you have a video and you need to realize what is that about or you need a specialist to peruse a line of record for you which is a picture of text and isn't in content organization.

Utilizing Bidirectional LSTMs, you feed the learning calculation with the first information once from starting as far as possible and once from end to starting. There are banters here however it typically learns quicker than a one-directional methodology in spite of the fact that it relies upon the assignment.

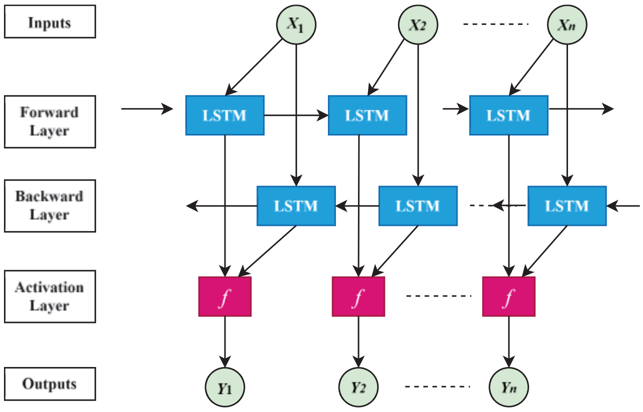

Fig. 3. A Bi-LSTM model applied on text data

C. Ensemble Learning

Ensemble Learning [26][27] is tied in with figuring out how to best join forecasts from various existing models (called the base-students). Each individual from the ensemble makes a commitment to the last yield and individual shortcomings are counterbalanced by the commitment of different individuals. The joined scholarly model is named the meta-learner. Ensemble learning [27] is executed to improve the exhibition of a model as far as forecast, order, work estimate, and so on. In this paper two ensemble approaches are used. Stack Level Stacking as shown in figure 4 and Weighted Average Ensemble as shown in figure 5.

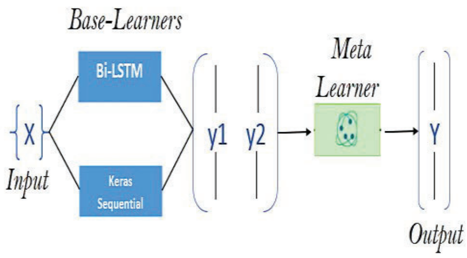

Fig. 4. Single Level Stacking Ensemble Model

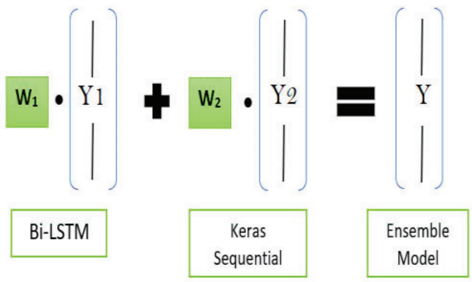

Fig. 5. Weighted Average Ensemble Model

VI. Results

We have used the two datasets mentioned above and the proposed ensemble methods performed very fairly for the given datasets. The weighted average ensemble gave an roc_auc score of 0.96 for the IMDB Reviews Dataset and a score of 0.91 for the Yelp Reviews Dataset. The Stack ensemble method gave a roc_auc score of 0.92 for the IMDB Reviews Dataset and a score of 0.90 for the Yelp Reviews Dataset as shown in Table I.

TABLE I. COMPARISON OF DIFFERENT CLASSIFICATION MODELS ON THE EVALUATION METRICS

Techniques	F1-score	Positive Class F1-score	Negative Class F1-score	Roc_auc Score
IMDB				
Bi-LSTM	0.89	0.89	0.89	0.88
Keras Sequential CNN	0.89	0.90	0.89	0.89
Weighted Average Ensemble	0.92	0.93	0.92	**0.96**
Stack Ensemble	0.90	0.91	0.92	**0.92**
YELP				
Bi-LSTM	0.81	0.86	0.71	0.80
Keras Sequential CNN	0.86	0.90	0.77	0.83
Weighted Average Ensemble	0.89	0.92	0.79	**0.91**
Stack Ensemble	0.90	0.91	0.78	**0.90**

Figure 6 shows the comparison of the ensemble methods with result obtained from the deep learning models. The results show that the ensemble method performs better than the individual Bi-LSTM and Keras Sequential CNN model on different types of datasets. Out of both ensemble approach, weighted average performs better than stack ensemble. Figure 7 shows the F1 Score comparison and it shows that weighted average works better than stack ensemble.

VII. Conclusion

In the field of textual sentiment analysis, the most significant approach is to compare different types of models and classifiers and then selecting the best amongst them to implement sentiment analysis. The ensemble modelling techniques have now been quite popular and been used in many areas to solve and improve the classification problem, but in the case of text sentiment analysis, less work has been done on the use of ensemble models. In our methodology, we used two ensemble models namely weighted average ensemble and stack ensemble which are shown to improve the performance of text sentiment analysis. The ensemble model is formed by different deep learning models like Bi-LSTM and Keras Sequential CNN model and the ensemble methods perform better than the standalone models. As future work, our main ideology would be to focus on the neutral sentiments i.e., neither negative nor positive and also applying our model to analyze the data on other social networking platforms.

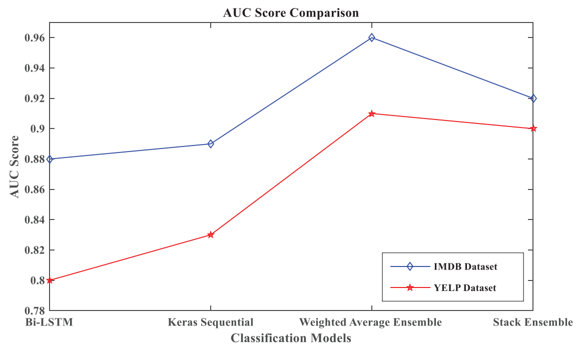

Fig. 6. AUC Score Comparison of different models

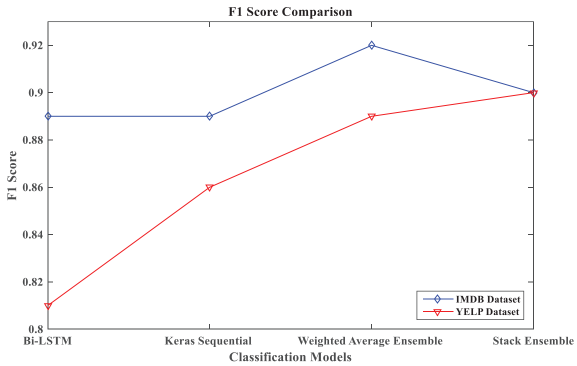

Fig. 7. F1 Score Comparison of different models

References:

[1] B. Liu, Sentiment Analysis: Mining Opinions, Sentiments, and Emotions. Cambridge University Press, 2015.

[2] A. Pak and P. Paroubek, "Twitter as a corpus for sen-timent analysis and opinion mining," in Proceedings of International Conference on Language Resources and Evaluation (LREC), 2010, pp. 17–23.

[3] Y. Kim, "Convolutional neural networks for sentence classification," in Proceedings of International Conference on Empirical Methods in Natural Language (EMNLP), 2014, pp. 1746–1751.

[4] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," Computational Linguistics, vol. 37, no. 2, pp. 267–307, 2011.

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[6] W. Zhao, Z. Guan, L. Chen, X. He, D. Cai, B. Wang, and Q. Wang, "Weakly-supervised deep embedding for product review sentiment analysis." IEEE Trans. Knowl. Data Eng., vol. 30, no. 1, pp. 185–197, 2018.

[7] M. Hu and B. Liu, "Mining and summarizing customer reviews," in Proceedings of ACM SIGKDD International Conference Knowledge Discovery Data Mining (KDD), 2004, pp. 168–177.

[8] T. Mullen and N. Collier, "Sentiment analysis using support vector machines with diverse information sources," in Proceedings of International Conference on Empirical Methods in Natural Language (EMNLP), 2004, pp. 412–418.

[9] G. Paltoglou and M. Thelwall, "A study of information retrieval weighting schemes for sentiment analysis," in Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL), 2010, pp. 1386–1395.

[10] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: A deep learning approach," in Proceedings of International Conference on Machine Learning (ICML), 2011, pp. 513–520.

[11] D. Tang, B. Qin, X. Feng, and T. Liu, "Effective lstms for target dependent sentiment classification," in Proceedings of International Conference on Computational Linguistics (COLING), 2016, pp. 3298–3307.

[12] A. E. Baccianella, Stefano, and F. Sebastiani, "Senti-wordnet 3.0: An enhanced lexical resource for sentiment analy-sis and opinion mining,"

in Proceedings of International Conference on Language Resources and Evaluation (LREC), 2010, pp. 2200–2204.

[13] https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews.

[14] https://www.kaggle.com/yelp-dataset/yelp-dataset.

[15] Classification using BERT” Department of Electronics and Computer Engineering Pulchowk Campus, Institute of Engineering, Tribhuvan University Lalitpur, Nepal.

[16] Victor SANH, Lysandre DEBUT, Julien CHAUMOND, Thomas WOLF “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter”

[17] LAHIRU SENEVIRATHNE, PIYUMAL DEMOTTE, BINOD KARUNANAYAKE, UDYOGI MUNASINGHE, and SURANGIKA RANATHUNGA ”Sentiment Analysis for Sinhala Language using Deep Learning Techniques”, Department of Computer Science and Engineering, University of Moratuwa.

[18] Lemmens, J., Burtenshaw, B., Lotfi, E., Markov, I., & Daelemans, W. (2020, July). Sarcasm detection using an ensemble approach. In Proceedings of the Second Workshop on Figurative Language Processing (pp. 264-269).

[19] Shervin Minaee , Elham Azimi , AmirAli Abdolrashidi “Deep-Sentiment: Sentiment Analysis Using Ensemble of CNN and Bi-LSTM Models”, New York University University of California, Riverside.

[20] Ayu Purwarianti , Ida Ayu Putu Ari Crisdayanti “Improving Bi-LSTM Performance for Indonesian Sentiment Analysis Using Paragraph Vector”, School of Electrical Engineering and Informatics Institut Teknologi Bandung Bandung, Indonesia.

[21] Yuxiao Chen, Jianbo Yuan, Quanzeng, Jiebo Luo “Twitter Sentiment Analysis via Bi-sense Emoji Embedding and Attention-based LSTM”.

[22] https://www.researchgate.net/profile/Paolo_Dellaversana“DEEP NEURAL NETWORKS WITH KERAS”.

[23] Qizhe Xie1 , Zihang Dai1 , Eduard Hovy , Minh-Thang Luong , Quoc V. Le “ Unsupervised Data Augmentation for Consistency Training”, Google Research, Brain Team, Carnegie Mellon University.

[24] Rie Johnson,Tong Zhang “Effective Use of Word Order for Text Categorization with Convolutional Neural Networks”.

[25] Bryan McCann, James Bradbury, Caiming Xiong, Richard Socher “Learned in Translation: Contextualized Word Vectors”.

[26] Liu, P., Chen, W., Ou, G., Wang, T., Yang, D., & Lei, K. (2014, June). Sarcasm detection in social media based on imbalanced classification. In International Conference on Web-Age Information Management (pp. 459-471). Springer, Cham.

[27] Lemmens, J., Burtenshaw, B., Lotfi, E., Markov, I., & Daelemans, W. (2020, July). Sarcasm detection using an ensemble approach. In Proceedings of the Second Workshop on Figurative Language Processing (pp. 264-269).