Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/5/20230541

Text classification algorithms exploration on sentiment analysis

Xupeng Zhang

University of Illinois at Urbana Champaign, Apt 309, 310 S 1st St, Champaign, Illinois

xupengz2@illinois.edu

Abstract. Internet provides us with an abundance of useful tools and data. However, it also generates a vast quantity of data that may bewilder us. There must be a technique for automatically processing these data. Here, text classification becomes useful. Text classification is the algorithm-based process of categorizing data inputs into distinct labels. For instance, email software utilizes it to assess if an email should be filtered into the spam folder, social media forums use it to classify postings into labels that are relevant to the topic, etc. Text categorization is utilized in a variety of businesses, including search engines, sentiment analysis, emergency response systems, chatbots, etc. Review websites have emerged in recent years where customers may share their opinions on a business or a product. The review is extremely emotive but crucial to the company. It is possible to accurately assess the reviews for the sentiment they present through text classification. This paper compares the efficacy of various text classification algorithms for sentiment analysis.

Keywords: Text Classification, Tokenization, Deep Learning, Sentiment Analysis.

1. Introduction

Costumer reviews, social network content is an important part in our daily lives. Natural language processing is a vital tool for us to categorize data generated in these platforms quickly and accurately. NLP research has a lengthy history. Recently, NLP has been a controversial method which boosts deep learning efficiency a lot. It categorizes data information in a fast and accurate way using deep learning algorithms. The goal of sentiment analysis is to identify expressions positive, negative, or neutral sentiments. Efficiency and accuracy are most important indicators of which algorithm is better. Making comparisons between different algorithms helps us identify a better way to apply on text classification.

Most text classification research focus on two components of machine-based sentiment analysis are standard models and deep learning [1]. The maximum entropy classifier, the naive Bayes classifier, and support vector machines are all examples of classic methods [2]. The deep learning methods are an emerging way to classify text data which utilize neural network algorithms. To achieve the goal of quickly processing and categorizing data, numerous actions must be taken.

This research study focuses on sentiment analysis using deep leaning methods and make comparisons among them through experiments.

DOI: 10.54254/2755-2721/5/20230541

2. Data Preprocessing

2.1. Splitting Data.

Social media is an important source of data for sentimental analysis. The information brought by social media is usually more complex [3]. Thus, the paper will choose the imdb movie review dataset [4] to work on. It contains 50,000 supervised movie reviews, consisting of half for training and half for validation. The text data is in string format and the label is in integer format. First, the data needs to be imported and split into training and validation groups in a ratio of 8:2. Shuffle method is applied to split the data to make sure the data is split without bias.

2.2. Noise Removal

Second, the data needs to be preprocessed so running deep learning algorithms on it is possible. For IMDB dataset, apply HTML decoding, stop words method, lowercasing [5]. However, applying noise reduction has drawbacks. When applying lowercasing to the text, it could potentially confound the machine to distinguish the text. After applying lowercasing, the word "US" in "English" will become "us," preventing the computer from recognizing the term and classifying the text into labels.

2.3. Tokenization

Tokenization is the step that split long paragraphs and sentences into smaller chunks which will be easier to analyze.

2.4. Stopword

Stop words are those that are frequently ignored when analyzing natural language. Before training, its likely to omit any irrelevant or deceptive information from the text. This will significantly increase how effectively text categorization work.

2.5. Word Embedding

Word embedding maps each word to a vector of real values for the purposes of language modeling and feature learning, so that words with similar meanings have comparable representations. [6].

3. Model Comparisons

3.1. Convolutional Neural Network

In a convolutional neural network, the convolution layer and the pooling layer are often the two most important components [7], as shown in Fig.1. Apply a convolution filter to the vectorized data before multiplying the values of the input matrix by the corresponding values in the convolutional filter. The pooling layer is similar to the convolutional layer. The max-pooling select the max value in the input feature map. The output map is significantly reduced in size which reduced the time needed to train the model.

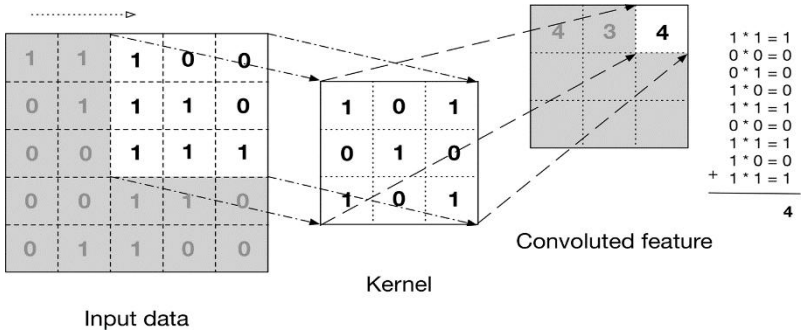

Figure 1. Image illustration of Convolution operation [8].

DOI: 10.54254/2755-2721/5/20230541

3.2. Multi-Layer Perceptron

At a minimum, a Multi-Layer Perceptron algorithm will have the following components: an input layer, a hidden layer, and an output layer, as shown in Fig. 2. Activation functions are typically included with an MLP package. Sigmoid activation is the most prevalent form.

$$y(V_i) = \tanh(vi) \text{ and } y(V_i) = \left((1 + e^{-v_i})^{-1}\right) \quad (1)$$

Layer (type)	Output Shape	Param #
embedding_3 (Embedding)	(None, 500, 32)	640000
flatten_3 (Flatten)	(None, 16000)	0
dense_6 (Dense)	(None, 250)	4000250
dense_7 (Dense)	(None, 1)	251

Figure 2. MLP architecture.

3.3. Bi-directional Long-Short Term Memory (BI-LSTM)

Through the use of bidirectional recurrent neural networks, the model is able to maintain both backward and forward knowledge of the sequence across all of the phases, as shown in Fig. 3. When you use Bi-LSTM, your inputs will be processed in two stages: the first will move in a reverse direction, and the second will move in a forward direction [9].

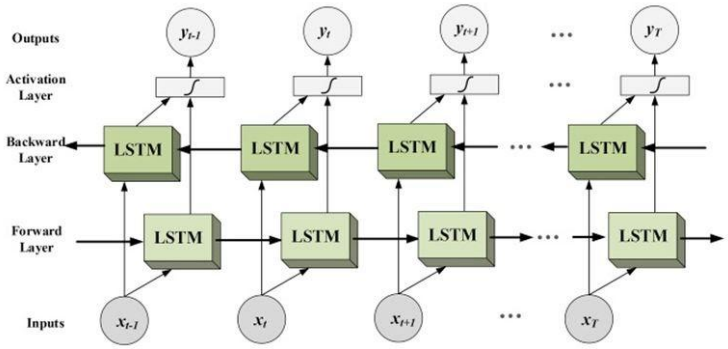

Figure 3. Bi-LSTM structure source.

4. Optimizer, Loss Function and Kernel Size Selection

While analyzing the performance of various algorithms on the same dataset, comparisons among different algorithms are made. The Adam optimizer was utilized for each of the models. The Imdb collection of movie reviews solely contains positive and negative opinions. As a loss function, thus, binary cross entropy should be utilized. It is difficult to find a balance between the kernel sizes of various models, as each operate differently. In the research paper [10], the author shows how kernel size can affect model training in different ways. In order to ensure a fair and accurate comparison, I specified the identical model dimensions and kernel size for each model.

5. Result

Through accuracy and loss curves, the performances of CNN, MLP, and Bi-LSTM on the sentiment analysis of imdb movie reviews are demonstrated. The average precision and training speed of several models are compared as well.

DOI: 10.54254/2755-2721/5/20230541

5.1. Accuracy and Loss curves

The accuracy and loss curves for the three algorithms stated above are shown in Figure 4.

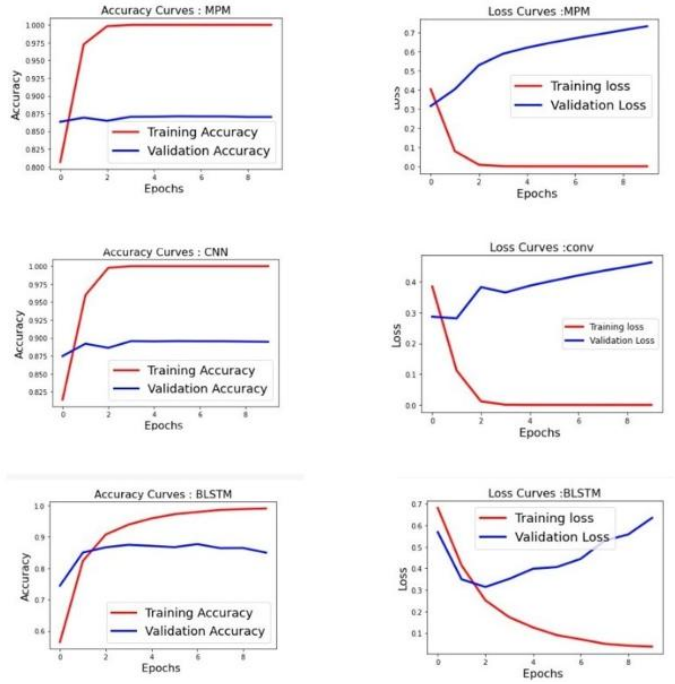

Figure 4. Accuracy and Loss Curves of Three Algorithms.

5.2. Mean Accuracy and Training Speed Comparisons

Table 1. Test accuracy score (%) and Training speed Comparisons (best results in bold).

	CNN	MLP	Bi-LSTM
Accuracy	**0.8952**	0.8517	0.8331
Speed	**14.2ms/step**	65.3ms/step	78ms/step

The results presented in Table 1 illustrate that the convolutional neural network achieves the highest levels of accuracy and training speed, whereas Bi-LSTM performs the poorest overall. Specifically, CNN reach state-of-the art performances on the imdb movie review dataset. The training speed of Bi-LSTM is expected since it performs forward and backward calculation during each step.

6. Conclusion

In this study, an overview of various text classification algorithms based on machine learning was presented. The paper contains information regarding the preprocessing and model architecture. The paper's conclusion by presenting an accuracy and training speed table that indicates which algorithms are superior at sentiment analysis is made.

Reference'

[1] Zhang, X and Zheng X 2016 Comparison of Text Sentiment Analysis Based on Machine Learning ISPDC pp. 230–233.
[2] Lai, Y 2019 A Comparison of Traditional Machine Learning and Deep Learning in Image

DOI: 10.54254/2755-2721/5/20230541

Recognition Journal of Physics: Conference Series IOP Publishing vol. 1314
[3] Dang N.C. 2020 Sentiment Analysis Based on Deep Learning: A Comparative Study Electronics, vol. 9 (MDPI AG) p 483
[4] Maas A.L., Daly R.E., Pham P.T., Huang D., Ng A., and Potts C 2011.Learning Word Vectors for Sentiment Analysis. ACL.
[5] Al Sharou K., Li Z., and Specia L 2021 Towards a better understanding of noise in Natural Language Processing. Proceedings of the Conference Recent Advances in Natural Language Processing - Deep Learning for Natural Language Processing Methods and Applications
[6] Incitti F 2022 Beyond Word Embeddings: A Survey Information Fusion, vol. 89 (Elsevier BV) pp. 418–36.
[7] Gasparetto A 2022 A Survey on Text Classification Algorithms: From Text to PredictionsInformation vol. 13 p.82
[8] Josh P, Gibson A 2017 Deep Learning: A Practitioner's Approach
[9] Jang B 2020 Bi-LSTM Model to Increase Accuracy in Text Classification: Combining Word2vec CNN and Attention Mechanism Applied Sciences, vol. 10 p. 5841
[10] Sood S, Sigh H 2022 Effect of Kernel Size in Deep Learning-Based Convolutional Neural Networks for Image Classification ECS Trans. 1078877