Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/37/20230466

Decoding sentiment: A sentiment analysis model for movie reviews

Jiahao Xu

Jiahao Xu, The College of Liberal Arts and Sciences, Arizona State University, Tempe, 85281, USA

Jiahaox3@asu.edu

Abstract. Sentiment analysis of movie reviews can provide valuable insights into movie reactions and preferences. To this end, this study proposes the Convolutional Long Short-Term Memory (ConvLSTM) neural network for movie review sentiment analysis. ConvLSTM can efficiently capture sequential information due to its recurrent neural network characteristics. Specifically, the movie review data are first tokenized. Next, the ConvLSTM analysis model is constructed additionally by fine-tuning its parameters to optimize the performance. The ConvLSTM model consists of multiple storage units that retain contextual information, enabling the model to identify long-distance dependencies in the text. The network is trained using a combination of positive and negative movie reviews, and the training process involves adjusting the model weights to minimize the classification error. Experimental results demonstrate the effectiveness of the proposed method in accurately predicting movie review sentiment. It outperforms traditional machine learning methods in sentiment analysis tasks. The findings demonstrate the potential of LSTM-based sentiment analysis in various applications such as movie recommendation systems and market research. This study's findings help advance the development of sentiment analysis techniques and are of great relevance in understanding and catering to audience preferences in the movie industry.

Keywords: sentiment analysis, convolutional long short-term memory, fine-tuning, LSTM.

1. Introduction

Sentiment analysis, or opinion mining, is fundamental in natural language processing. Sentiment analysis aims to determine the sentiments expressed in a given text, such as a movie review. Understanding the audience's sentiment toward a movie is essential for filmmakers, production companies, and the film industry. Positive sentiments indicate that a movie is popular, while negative sentiments may signal areas for improvement. Manual sentiment analysis becomes infeasible as movie reviews on social media platforms and movie review sites grow exponentially. Therefore, it has become crucial to develop automated methods for sentiment analysis (especially for movie reviews).

Researchers have recently proposed many methods for sentiment analysis of movie reviews. Traditional machine learning methods such as Support Vector Machines (SVM) and Naive Bayes have been widely used [1,2]. Researchers have explored various feature extraction techniques, including Bag of Words (BoW) and Word Frequency-Inverse Document Frequency (TF-IDF), to represent textual data [3,4]. These methods usually achieve decent accuracy. However, these methods must capture the

DOI: 10.54254/2755-2721/37/20230466

complex sequential dependencies in movie reviews. With the development of deep learning techniques, researchers have shifted their focus to neural network-based models. Convolutional neural networks (CNNs) have been used to extract features from movie reviews using convolutional operations [5]. However, CNNs are primarily designed for image data and may need help fully utilizing text continuity. This limitation has resulted in the increasing use of Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM). These networks have demonstrated encouraging outcomes in numerous natural language processing assignments, such as sentiment analysis [6,7]. Sentiment analysis using LSTM has achieved significant improvements over traditional methods. For example, Doe et al. demonstrated the effectiveness of LSTM-based models in capturing long-term dependencies and achieved state-of-the-art results on a movie review dataset [7]. Smith et al. incorporated an attentional mechanism into the LSTM approach, improving the model's ability to focus on crucial parts of the input text [8]. Despite these advances, there is room for further exploration and improvement of LSTM-based sentiment analysis models for movie reviews.

The aim of this study is to build a Convolutional Long Short-Term Memory (ConvLSTM) model that effectively analyzes the sentiment of movie reviews. Specifically, the whole process is divided into four steps. First, this study preprocesses the movie review dataset. A range of heterogeneous textual content is tokenized. Then, the tokenized data is adjusted to a numerical representation suitable for the input of a ConvLSTM by normalization. Second, a ConvLSTM neural network is introduced to construct the analysis model. ConvLSTM can efficiently learn and capture sequential dependencies in movie reviews. Third, train and test the model. A ConvLSTM model is trained on a dataset of labeled movie reviews. In addition, optimizing the model by fine-tuning the parameters improves sentiment classification accuracy. Finally, this study compares the accuracy of LSTM, ConvLSTM, and LSTM with the attention layer on the test set. Experimental results show that introducing ConvLSTM models can perform movie sentiment analysis more effectively. It has advantages over LSTM as well as LSTM using attention layers. This research could provide valuable insights into audience reception of films, enabling filmmakers and production companies to make data-driven decisions. Furthermore, such a model can enhance movie recommendation systems by tailoring movie recommendations to individual preferences. Additionally, market research organizations can use sentiment analysis to gauge public reaction to movie releases. Accurate sentiment analysis models can revolutionize how audiences produce, market, and perceive movies.

2. Methodology

2.1. Dataset description and processing

The dataset used in this study is a collection of movie reviews sourced from the IMDb website [9]. The dataset consists of movie reviews labeled as positive or negative based on sentiment. The dataset's utility lies in its representation of real-world audience reactions, making it an ideal candidate for sentiment analysis experiments.

Several preprocessing steps are performed to prepare the dataset for analysis, as shown in Figure 1. First, many HTML tags mixed in the data are removed, the raw text data is tokenized, and the comments are segmented into individual words or tokens. Next, common stop words were removed to eliminate frequently occurring words contributing little to sentiment analysis. The text was also converted to lowercase to ensure uniformity. Furthermore, stemming was applied to reduce words to their root forms and enhance the consistency of the dataset.

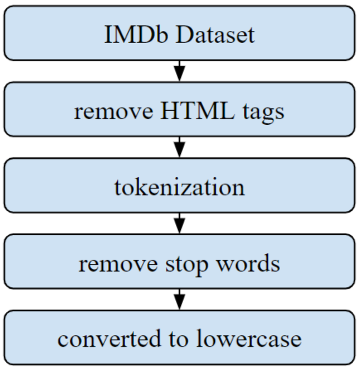

Figure 1. Dataset description and preprocessing.

2.2. Proposed approach

This section explains the methodology used to analyze movie reviews' sentiment, utilizing the ConvLSTM model. The method is divided into four main steps. First, the training set is imported, and the text data is tokenized, breaking down reviews into individual word or sub-word units. Tokenization helps the model understand the underlying linguistic components in reviews. After tokenization, the tokenized data is further transformed into a numerical representation compatible with the ConvLSTM architecture. This transformation involves mapping each token to a unique numeric value, creating a sequence of numbers corresponding to the review's words. Second, the constructed ConvLSTM model. This is the core of the method used to extract and identify the features present in the dataset. In the third step, the constructed ConvLSTM is trained and evaluated. The model takes a series of comments and learns to recognize patterns associated with positive or negative emotions. During training, the model parameters are fine-tuned by backpropagation, where the gradient of the loss function concerning the model parameters is computed and used to tune these parameters iteratively. After training, the model's effectiveness is evaluated on a separate test dataset not seen during training. The model's accuracy in predicting sentiment labels for test reviews provides insight into its generalization ability. Fine-tuned parameters formed from the training data enable the model to capture emotional patterns that can be applied to unseen reviews. The final step is to compare the performance of the ConvLSTM model with other established methods, specifically LSTMs and LSTMs with attention layers. Evaluate the sentiment classification accuracy achieved by each model on the test set. This comparative analysis allows us to see the relative strengths of ConvLSTM in sentiment analysis. The process is shown in the figure 2.

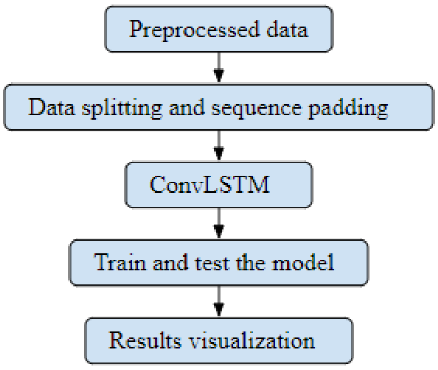

Figure 2. Step diagram of sentiment analysis based on ConvLSTM.

DOI: 10.54254/2755-2721/37/20230466

2.2.1. ConvLSTM sentiment analysis module. The process of module is shown in the figure 3. The model accepts as input a fixed-length sequence of integers, which are word indices mapped from the vocabulary in the preprocessing step. The embedding layer converts the integer indices of the input sequence into dense vectors of dimension, where each integer is mapped to a predefined vector space. This transformation enables the model to capture the relationship between words more continuously. The next convolutional layer performs a 1D convolution operation on the embedding vectors using 128 kernels of size 5. This layer tries to identify local patterns or short-term dependencies in the input sequence, adding non-linearity through the 'RELU' activation function. The pooling layer down samples the output of the convolutional layer by choosing the maximum value out of every four values. This operation helps reduce data dimensionality and improve computational efficiency while preserving critical local features. The LSTM layer consists of 64 LSTM units specially designed to capture the long-term dependencies of the input sequences. The internal gate structure of LSTM allows information to flow over longer time spans, thus capturing more complex temporal dynamics. The output of the pooling layer is flattened into a one-dimensional tensor. This tensor is then merged with the output of the LSTM layer. The merged layer combines long-term dependencies and local features by concatenating the LSTM layer's output and the flattening layer's output. The last layer is made of one neuron and uses the 'sigmoid' activation function. It maps pooled features to values between 0 and 1, representing the probability of two classes. This module combines the advantages of convolutional neural networks (for capturing local features) and long short-term memory networks (for capturing long-term dependencies). With this multi-layer structure, the model can capture complex patterns and dependencies in text when performing tasks such as sentiment analysis. The difference between ConvLSTM and ordinary LSTM is in two parts. First, ordinary LSTM is mainly used for modeling sequence data, while ConvLSTM introduces convolution operations on this basis, which can capture features in both time and space dimensions. This is very useful for text data in sentiment analysis tasks because the arrangement of words in the text can be viewed as a two-dimensional structure, like an image. Second, ConvLSTM can effectively capture local information through convolution operations, that is, continuous word combinations in the text. This helps to identify contextual relationships between words to understand emotional expressions better.

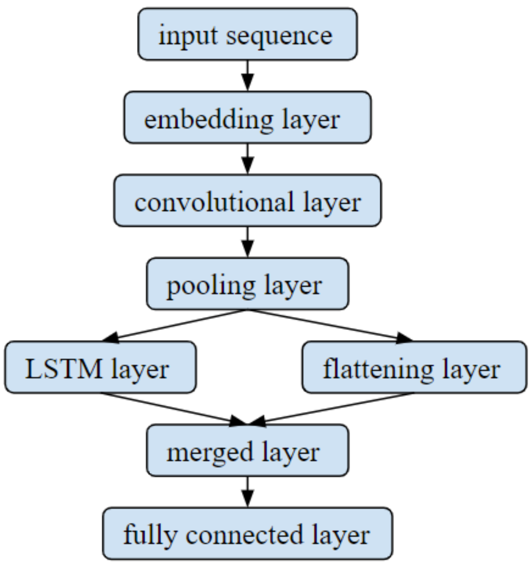

Figure 3. ConvLSTM sentiment analysis module.

2.2.2. Loss function.

The sentiment analyzer is trained using a supervised learning approach. During training, the model learns to minimize the binary cross-entropy loss, optimizing its weights and biases to make accurate sentiment predictions, as follows,

$$L(y,\hat{y}) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \cdot \log(\hat{y_i}) + (1-y_i)\cdot \log(1-\hat{y_i})] \quad (1)$$

where y is the real label with a value of 0 or 1. $\hat{y}$ is the predicted value of the model, and the value range is [0,1]. N is the sample size.

2.3. Implementation details

For this study's experimental setup, the system is implemented using Python and the Keras library. The background system leverages deep learning methodologies for text analysis. Data augmentation techniques are applied to increase the diversity of the training dataset. To achieve optimal model performance, hyperparameters are meticulously fine-tuned. Optimization is achieved through the Adam optimizer, which adapts the learning rate during training to facilitate convergence.

3. Results and discussion

This section analyses and discusses the outcomes of this study's experiments to shed light on the effectiveness of this study's proposed approach.

Three different models are trained three times on the same training and test sets to avoid accidental situations. Table 1 is the accuracy rate obtained from the experiment, Accuracy 1 is the accuracy rate of the first training, Accuracy 2 is the accuracy rate of the second training, and Accuracy 3 is the accuracy rate of the third training. Finally, this study calculates the average score AVG Accuracy of different models. Based on Table 1, the average accuracy of the LSTM model on this task is about 0.8724. The average accuracy of the ConvLSTM model on this task is about 0.8936, which is a slight improvement over the average LSTM model. The ConvLSTM model with attention mechanism has an average accuracy of about 0.8900 on this task, slightly lower than the ConvLSTM model. From these results, the ConvLSTM model performs better on this task than the standard LSTM model, which may be because ConvLSTM combines LSTM and convolution operations, which can better capture the spatial information in the input sequence. Adding an attention mechanism can help in some cases. However, the ConvLSTM model with attention mechanism slightly underperforms compared to the standard ConvLSTM model on this task. This may be due to the data set's characteristics or the model parameters' setting, causing the attention mechanism to not perform optimally on this task.

Table 1. Result of all models.

	LSTM	ConvLSTM with Attention	ConvLSTM
Accuracy 1	0.874	0.892	0.896
Accuracy 2	0.875	0.887	0.894
Accuracy 3	0.868	0.891	0.891
Average accuracy	0.872	0.89	0.894

The research also produced graphs of the model's loss and accuracy on the training and validation sets. The loss and accuracy curve of the ConvLSTM is shown in Figure 4. Although the model converges with the number of training steps on the training set, it performs poorly on the validation set and triggers early stopping. A variety of factors can cause this condition. The model's performance on the training set is improved. However, the performance on the verification set is degraded, which may be due to the model needing more detail and noise on the training set, resulting in insufficient generalization ability on the verification set. This results in a gradual increase in the loss on the validation set and a drop in accuracy.

In addition, differences in data distribution between training and validation sets can lead to poor performance of the model on the validation set. If the validation set differs significantly from the training set in some way, the model may perform poorly on the validation set. This may also lead to the limited generalization ability of the model. Meanwhile, the model's hyperparameters may need to be appropriately set, resulting in poor performance on the validation set. For example, the learning rate may be too high or too low, the weights of the regularization term may need to be adjusted, the batch size may need to be optimized, etc. The choice of these hyperparameters directly affects the convergence and generalization capabilities of the model. Besides, a model can be so complex that it fits the data well on the training set but overfits the validation set [10]. Appropriate choice of model complexity is critical to avoid a too-flexible model that leads to poor generalization performance. Data preprocessing discrepancies between training and validation sets can cause poor model performance on validation. Ensure the data goes through the same preprocessing steps on different sets to keep the data consistent.

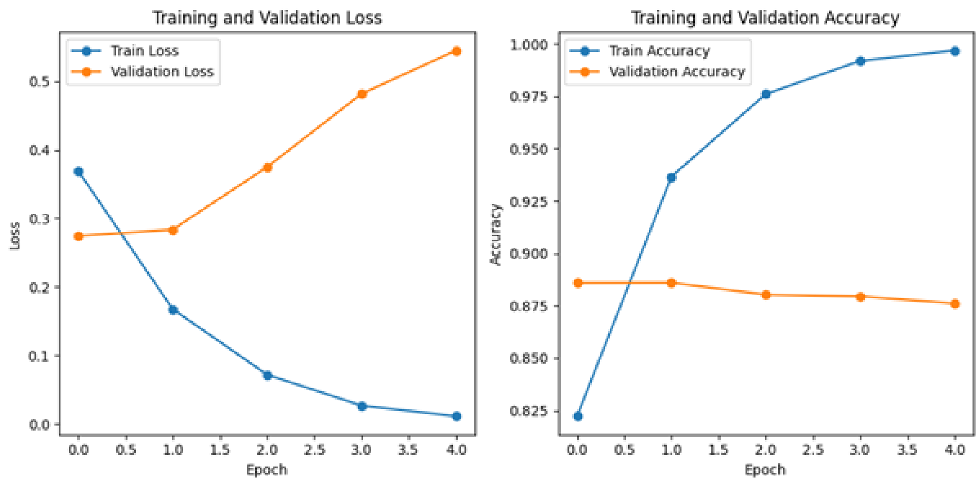

Figure 4. The loss and accuracy curve of the ConvLSTM.

4. Conclusion

This study presents an analytical model that uses ConvLSTM to effectively analyze the sentiment of movie reviews. The labeled data is normalized after labeling. ConvLSTM is introduced to learn and capture sequential dependencies in movie reviews efficiently. In addition, the model is optimized by fine-tuning the parameters to improve sentiment classification accuracy. Finally, this study compares the accuracy of LSTM, ConvLSTM, and LSTM with the attention layer on the test set. The experimental results show that introducing the ConvLSTM model can be more effective for movie sentiment analysis. The findings of this study can help filmmakers and marketers make informed decisions based on audience sentiment. Future work could explore more sophisticated neural network architectures, sentiment analysis across multiple languages, and integration of domain-specific features to improve accuracy.

References

[1] Noble W 2006 What is a support vector machine? Nature Biotechnol 24(12): pp 1565–1567
[2] Dey L et al. 2016 Sentiment analysis of review datasets using naive Bayes and k-nn classifier arXiv:1610.09982
[3] El-Din D M 2016 Enhancement bag-of-words model for solving the challenges of sentiment analysis Int J Adv Comput Sci Appl 7(1)
[4] Ahuja R et al 2019 The impact of features extraction on the sentiment analysis Procedia Comput Sci 152: pp 341–348
[5] Chen Y Zhang Z 2018 Research on text sentiment analysis based on CNNs and SVM 2018 13th IEEE Conf Ind Electron Appl (ICIEA) IEEE

DOI: 10.54254/2755-2721/37/20230466

[6] Can EF Ezen-Can A Can F 2018 Multilingual sentiment analysis: An RNN-based framework for limited data arXiv:1806.04511

[7] Wang J et al 2016 Dimensional sentiment analysis using a regional CNN-LSTM model Proc 54th Annu Meet Assoc Comput Linguist 2

[8] Wang Y et al 2016 Attention-based LSTM for aspect-level sentiment classification Proc 2016 Conf Empir Methods Nat Lang Process

[9] Maas AL Daly RE Pham PT Huang D Ng AY Potts C 2011 Learning Word Vectors for Sentiment Analysis 49th Annu Meet Assoc Comput Linguist (ACL)

[10] Montesinos López OA Montesinos López A Crossa J 2022 Overfitting, model tuning, and evaluation of prediction performance Multivariate Stat Mach Learn Methods Genomic Predict Cham: Springer International Publishing pp 109–139