Aspect-Based Sentimental Analysis On Social Media Data Using Deep Learning Methods

Samson Ebenezar Uthirapathy[1], Domnic Sandanam[2]

u.samson@gmail.com[1], dominc@nitt.edu[2]

Saveetha School of Engineering, Saveetha Institute of Medical And Technical Sciences, Chennai, India1, Department of CA, National Institute of Technology, Tiruchirappalli, India2

Abstract. Nowadays, reviews related to the products given by users and experts in social media are significant references to help product manufacturers make better decisions about prompting items and placing them in the real world. So analysing aspects such as product-related sentiments is an important task. The target of aspect-based opinion analysis is to determine the emotional valence of each aspect phrase included inside a given sentence. However, the earlier models make the error of identifying irrelevant contextual terms as cues for determining aspect sentiment. They also overlook syntactical limitations and long-range sentiment dependencies. Hence, we assumed a model using LSTM with GCN to distinguish engagers' review opinions under various aspects with three different labels: negative, neutral, and positive. The proposed model LSTM+GCN shows better performance than other models in terms of 4% with precision, 4% with recall, and 5% with accuracy.

Keywords: LSTM, RNN, Aspect based Sentiment Analysis, Machine Learning.

1 Introduction

In today's digital world, people express their thoughts and opinions on platforms like Twitter, Facebook, and Instagram, making sentiment analysis crucial [1]. Aspect-based sentiment Analysis (ABSA) is a method used to classify sentiments related to specific aspects of text. It's valuable in contexts such as healthcare and product reviews. Organizations can use ABSA to analyze customer feedback, like medicine-related tweets, to understand patient conditions and make informed decisions about products. ABSA helps deal with the challenge of handling reviews with multiple aspects by categorizing sentiments at the aspect level.

Machine learning, including techniques like RNN and CNN, is commonly used for sentiment analysis. Deep learning methods are essential for understanding complex data patterns in text and images, enabling accurate predictions and insights. For usual sentimental analysis, there are primarily five steps [2] namely data preparation, reviews evaluation, sentimental classification, and outcomes which relegates the reviews into three varieties: positive, negative, and neutral.

IACIDS 2023, November 23-25, Lavasa, India

DOI 10.4108/eai.23-11-2023.2343146

This paper proposes a strategy to improve ASBA classification accuracy using an RNN method called LSTM with GCN. LSTM The paper's main subscriptions are as follows:

- Identifying the best feature extraction method to amend the model's classification accuracy.
- Designing an opinion classification model using the LSTM with GCN method to improve the accuracy of the classification model.

2 Related Work

This part explains some of the related research to our proposed work done in the recent past.
Soni and Mathur [3] have proposed an LSTM with an encoder attention mechanism for sentiment analysis here they used attention mechanisms to extract the aspect features and then go through LSTM for sentiment classification. Wadawadagi and Pagi [4] have presented a novel method that includes Bidirectional LSTM with conditional random field(CRF) and a Polarity Enriched Attention Neural (PEAN) model to extract aspect terms and categorization of sentiments. A novel approach for quantifying sentiment-orientated classification has been proposed by Gurunathan [5]. Divate [6] analyzed the sentiments present in the Marathi tweets using LSTM.
Uthirapathy and Sandanam [7] proposed a novel method using fuzzy c means closeting for sentiment classification in tweets. Ugochi and Prasad [8] developed a classification framework for sentiment analysis. Naive Bayes was utilized to predict the classes. It was contrasted with other techniques like SVM and KNN. A hierarchical model using LSTM for aspect-oriented sentiment analysis was discussed by Ruder et al. [9]. Liang et al. [10] developed a technique to identify and extract the aspect words and the opinion words using a dependency tree method called Senticnet along with a GCN. The unique improved graph model considers both contextual word-aspect word dependencies and opinion-aspect word affective information.
To ensure that the SemGCN component appropriately captures semantic relationships between words, Li et al.[11] suggested using orthogonal and differential regularizers to provide bounds on attention ratings. The orthogonal regularizer helps the SemGCN find more closely related phrases by lowering the amount of overlap between the words being learned. Karimi et al. [12] presented a novel approach based on Adversarial Training using BERT by applying adversarial learning to the two main goals in sentiment analysis: Aspect term extraction and Sentiment categorization (BAT). Hu et al.[13] proposed a Multi-level Semantic Relation-enhanced Learning Network (MSRL-Net) and offered as a model for ABSA, along with a self-supervised task for classifying pairs of phrases based on their semantic relationships.

3. Proposed Methodology

This section demonstrates the four basic components that make up the process of classifying the sentiments with the aspect features. Data gathering, pre-processing of data, extraction of features, aspect term extraction, and classification are the phases followed in the proposed methodology.

3.1 Data Pre-processing

There are words, figures, and special symbols in the reviews that need to be studied. Data pre-processing involves removing irrelevant single quotes, emails, newline characters, and background noise from the data. The entire text has been lowercase and all punctuation has been removed from the reviews or texts. The punctuation and numerals are eliminated since sentiment analysis is unaffected by these elements. Additionally, several stop words—commonly defined as words like "is off," "of," etc.—are also deleted. Then the other preprocessing techniques such as stemming, tokenization, and word embedding also performed with the input text reviews. For word embedding the methods TF-IDF and BERT [14]have been used.

3.2 Classification layer using LSTM and GCN

In the next step, the embedded input is forwarded to the LSTM[15] and GCN classification model. Consider the context term $c = \{ c_1, c_2, c_3 ,..., c_n\}$ and the aspect term e, the relative distance s_j is calculated between the j^{th} context word and the specific aspect term in a phrase. If the particular aspect term contains several different units, we will use the mean value of the word embeddings of the different units to determine the embedding value of the target aspect term. The time taken to pass is denoted by the numeral *n*. As inputs, LSTM+GCN takes into consideration both the overall environment as well as the particular component that is being targeted.

3.2.1 Context term and Aspect term embedding

It is necessary to convert the one-hot vector format as the word embedding before feeding the context and aspect target into LSTM+GCN layers. The word is transformed from a vector with high dimensions and dispersed points into a vector with low dimensions and concentrated points. By performing a mathematical operation in the embedding space, we can compute the compositionality.Given the matrix $M \in R^{V \times d}$ of the word embedding, where the vocabulary size is denoted by the letter *V*, and the letter *d* denotes the word embedding dimension. The following is an example of the dependency matrix that may be used to acquire context embedding c_j' from the lookup table using equation (1).

$$c_j' = lookup(c_j) = c_j^T . M \quad (1)$$

From the word embedding matrix M the aspect embedding e_j' can be obtained using equation (2).

$$e_j' = \frac{1}{r}\sum_j^r \quad lookup(e_j) = \frac{1}{r}\sum_j^r \quad e_j^T . M \quad (2)$$

The relative distance between the context word and the particular aspect term can be determined by the location embedding based on equation (3) if the aspect term contains one word.

$$s_j = |i^s - j^s| \quad (3)$$

Then the location embedding l_j can be calculated from the lookup table using equation (4).

$$l_j' = lookup(l_j) = l_j^T . M \quad (4)$$

Finally, the input vector is finalized by concatenating the location embedding of each word, context embedding, and aspect term embedding.

3.3.2 LSTM Layers

After that, the input vector is given to the LSTM layer to learn the semantic data of the sentences word by word as shown in Figure 1. The LSTM extends short-term memory and it consists of cells. Each cell has an output gate, a forget gate, and one or more input gates.

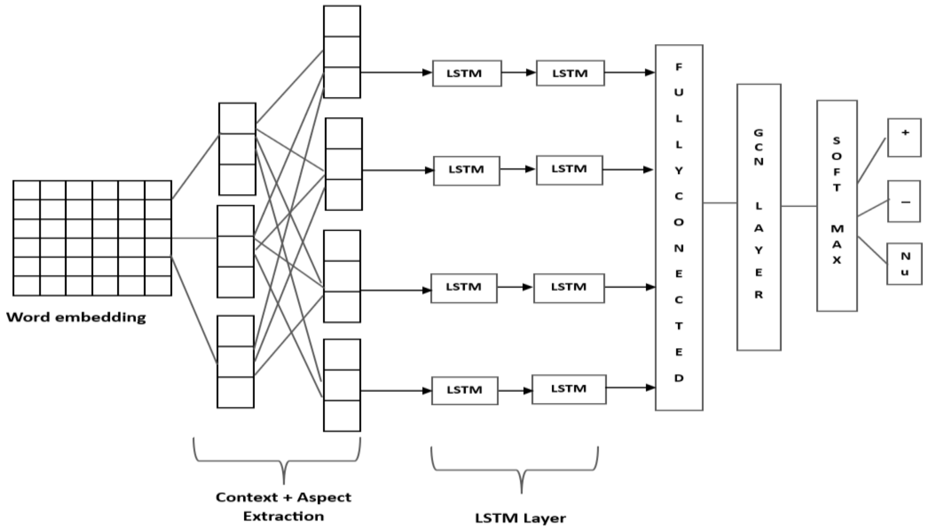


Fig. 1: LSTM+GCN Model

The LSTM hidden layer output is mentioned in equation (5).

$$h_t = o_t * tanh\ tanh\ (S^t) \qquad (5)$$

3.3.3 Graph convolutional networks

Before passing the h_t to the final softmax output layer, it is passed to the GCN layer, the GCN layer takes the dependency matrix as input. The GCN considers the sentence and calculates the dependency between the words using a convolutional graph. To construct a new dependency tree G with k nodes, here we consider sentence S with N words. In this case, the words are the nodes and the connections between any two words are the edges. It is therefore possible to reconstruct the dependency matrix from the LSTM layer output as $X_{ij} \in R^{k*k}$. For the dependent position, 1 is used and for the other positions 0 is used. h_i^{l-1} represents the input to the i^th node, while h_i^l represent the output. Equation (6) represents the output of the hidden layer in the graph convolutional neural network.

$$H^m = \sigma(\sum_{j=1}^{n} D_{ij} W^l h_i^{l-1} + b^l) \qquad (6)$$

The H^m is the aspect features extracted by the GCN that efficiently represent the aspect words in the sentences.

3.3.4 Output layer

The aspect features of sentence *S* are passed to the output layer for polarity classification as positive, neutral and negative. The activation function is Softmax in the output layer it takes the form as mentioned in equation (7).

$$y(a) = softmax\ (w_o S + b_0) \qquad (7)$$

3.3.5 Optimizer

In the proposed LSTM-GCN model, the Adam backpropagation algorithm is used to adjust the learning rate to advance the suggested model's accuracy. Adam is mathematically represented in the equation (8).

$$w_{t+1} = w_t - \alpha m_t \qquad (8)$$

$$m_t = \beta m_{t-1} + (1 - \beta) \left[\frac{\delta L}{\delta w_t}\right] \qquad (9)$$

In equations (8) and (9), m_t represents the aggregation of gradients at time t. m_{t-1} it is the gradient aggregation at time t-1. w_t, w_{t+1} are weights at time t and t-1 respectively. α is learning. β is the moving average parameter that takes the constant value of 0.9. δL is the loss function derivative at time t and δw_t is the weight derivative.

4 Results and Discussion

The suggested model LSTM+GCN is evaluated using the following two publicly available datasets from the Kaggle website[16]. In this work, two data sets have been employed to determine the efficiency of the proposed LSTM+GCN method for aspect-based sentiment categorization. The data sets contain reviews related to drugs. The health-related drug data set consists of approximately 30000 reviews and UCI ML drug review data set consists of 215000 reviews. With the use of statistics, we assessed the model using the following measures namely precision, recall, F-score and Accuracy.

Table 1: Parameter Settings

Parameters	Values
LSTM Layers	3
GCN Layer	1
Softmax Layer	1
Optimizer(Backpropagation)	rmsprof
Word embeddings	TF-IDF, BERT
Learning rate TF-IDF	1e -3
Learning rate BERT	2e -4
Dropout	0.5
Batch Size	128
Number of Ephocs	100

4.1 Performance Evolution

The outcomes were compared using four different approaches namely LSTM, BI-LSTM, GRU, and simple RNN.

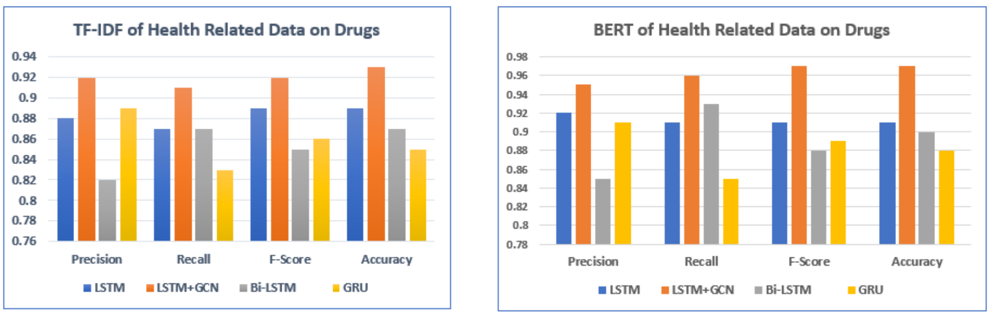
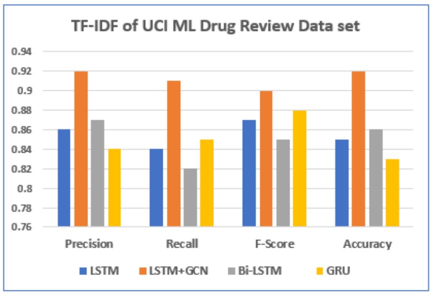


Fig 2: Performance comparison with Health-Related Data Set



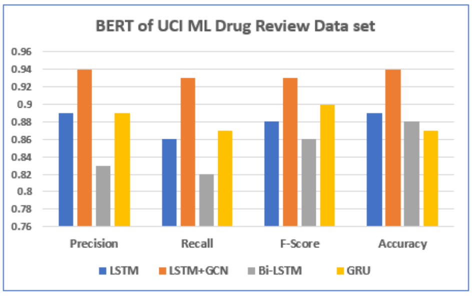


Fig 3: Performance comparison with UCI ML Drug Review Data Set

With the help of the experiment, the results are shown in Figures 2 and 3. The proposed LSTM+GCN model has produced better results in terms of classification with the BERT word embedding method.

5. Conclusion

This paper deals with two different data sets to validate the efficiency of the proposed LSTM+GCN model for aspect-based sentiment analysis. Here two different word embedding methods namely TF-IDF and BERT have been used. First LSTM layer is used for aspect term extraction, the GCN layer is used for sentiment polarity classification finally the SoftMax layer is used for normalizing the multiclass output. It categorized the participants' review sentences as good, neutral, or negative. The proposed LSTM+GCN model shows better results with BERT word embedding compared with TF-IDF word embedding. Finally, the proposed model was compared with other existing models in aspect-based sentiment classification, the results reveal that the proposed LSTM+GCN model performs better than other models in terms of 4% with precision, 4% with recall and 5% with accuracy.

References

[1] Li, J., Fong, S., Zhuang, Y., and Khoury, R., 2019. “Hierarchical classification in text mining for sentiment analysis”. In 2019 International Conference on Soft Computing and Machine-Intelligence, IEEE, pp. 46– 51.

[2]Uthirapathy, S.E. and Domnic, S., 2020. Real-Time Opinion Prediction Method for Emergency Public Events in Social Media Networks Using Opinion Hit Matrix. Rev. d'Intelligence Artif., 34(4), pp.507-514

[3] Soni, J., Mathur, K. Sentiment analysis based on aspect and context fusion using attention encoder with LSTM. Int. j. inf. tecnol. 14, 3611–3618 (2022).

[4] Wadawadagi, R. and Pagi, V., 2022. Polarity enriched attention network for aspect-based sentiment analysis. International Journal of Information Technology, 14(6), pp.2767-2778

[5] Gurunathan, P., 2022. Joint aspect-opinion extraction and sentiment orientation detection in university reviews. International Journal of Information Technology, 14(6), pp.3213-3225.

[6] Divate, M.S., 2021. Sentiment analysis of Marathi news using LSTM. International journal of Information technology, 13(5), pp.2069-2074.

[7] Uthirapathy, S.E. and Sandanam, D., 2023. Predicting opinion evolution based on information diffusion in social networks using a hybrid fuzzy based approach. International Journal of Information Technology, 15(1), pp.87-100

[8] Ugochi, O., Prasad, R., Odu, N., Ogidiaka, E. and Ibrahim, B.H., 2022. Customer opinion mining in electricity distribution company using Twitter topic modeling and logistic regression. International Journal of Information Technology, 14(4), pp.2005-2012.

[9] Ruder, S., Ghaffari, P., and Breslin, J. G., 2019. “A hierarchical model of reviews for aspect-based sentiment analysis”. arXiv preprint arXiv:1609.02745.
[10] Liang, B., Su, H., Gui, L., Cambria, E. and Xu, R., 2022. Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. Knowledge-Based Systems, 235, p.107643
[11] Li, R., Chen, H., Feng, F., Ma, Z., Wang, X. and Hovy, E., 2021, August. Dual graph convolutional networks for aspect-based sentiment analysis. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (pp. 6319-6329).
[12] Karimi, A., Rossi, L. and Prati, A., 2021, January. Adversarial training for aspect-based sentiment analysis with bert. In 2020 25th International Conference on Pattern Recognition (ICPR) (pp. 8797-8803). IEEE.
[13] Hu, Z., Wang, Z., Wang, Y. and Tan, A.H., 2023. MSRL-Net: A multi-level semantic relation-enhanced learning network for aspect-based sentiment analysis. Expert Systems with Applications, p.119492
[14] Xu, H., Liu, B., Shu, L., and Yu, P. S., 2019. “Bert post-training for review reading comprehension and aspect-based sentiment analysis”. arXiv preprint arXiv:1904.02232.
[15] Huang, B., Guo, R., Zhu, Y., Fang, Z., Zeng, G., Liu, J., Wang, Y., Fujita, H., and Shi, Z., 2022. “Aspect-level sentiment analysis with aspect-specific context position information”. Knowledge-Based Systems, 243, p. 108473.
[16] https://www.kaggle.com/datasets/jessicali9530/kuc-hackathon-winter-2018?select=drugsComTest_raw.csv