Enhancing IMDB Sentiment Analysis with GCN and Lexical Resources

A. Ali Husnain SN:6747279, ah530@uowmail.edu.au
B. Christy Natalia Jusman SN:7699736, cnj735@uowmail.edu.au
C. Huanfu Zhang SN:6307176, hz065@uowmail.edu.au
D. Lei Xie SN:7243984, lx978@uowmail.edu.au
School of Computing and Information Technology
University of Wollongong
CSCI933,

May 31, 2024

Group Contribution

Group member	Ali	Christy	Huanfu	Lei
Contribution mark	10.0	10.0	10.0	10.0

Abstract

In this report, we are experimenting with various methods of aspect-based sentiment analysis to predict the sentiment of IMDB movie reviews. The reviews are classified solely into positive or negative classes based on various aspects discussed by the reviewers. The goal of this experiment is to address language processing challenges and accurately predict the data. Three experiments are conducted in this report. First, we implemented a model named Bag-of-Words and convolutional neural network (CNN) model with embedding layer, which has been tested accurately for ABSA achieving 89% level of accuracy. Additionally, we proposed 2 new models of Graph Neural Network (GNN) and utilizing lexical resources. IN GNN, we represent each word as a graph and there will be nodes connected based on the words combination. However, the experiment does not work perfectly as it only achieves 50% level of accuracy. Surprisingly, with our model that utilizes lexical resources, it achieves 88.82% level of accuracy.

1 Introduction

In this modern digital era, Internet contains information that can be accessed freely, affecting our day-to-day decisions . A good application is making movie reviews; the web resources serves as a base where people search for reviews and insights before deciding which movie to watch. However, it might be overwhelming to read through all reviews on the internet and we need an instant way to do it.

The answer comes from Sentiment Analysis (SA). It takes a text, defines its expressed sentiment, analyzes it, and tries to compile opinions, identify the feelings they express, and classify their orientation (W. Medhat, 2014). However, SA is not enough since it is just analyzing the sentiment based on a text. It is assumed by the emphasis on the document or sentence level that there is only one issue stated in the document or phrase, which is not always the case. Therefore, in order to identify entities and related aspects and categorize feelings associated with these entities and aspects, a more comprehensive study is needed to extract the aspects being discussed which is known as Aspect-Based Sentiment Analysis (ABSA) (M. Hu, 2004). There are many methods provided by researchers to obtain the aspects of the text. First, it utilizes deep learning named Long Short-Term Memory (LSTM) neural network to do both aspect extraction and polarity identification (M. Al-Smadi, 2018). Additionally, another method is also proposed by M. E. Mowlaei and Keshavarz in 2018 using lexicon generation for the polarity identification of the text. The proposed strategy involves using a mechanism that produces scores to rate every word and place it within a phrase by identifying the nearest aspect term; then it adjusts the frequency based on how this term is annotated. However, there are many more methods proposed by other researchers to improve the accuracy of the ABSA.

2 Literature review

Sentiment analysis has evolved from basic classifications of 'positive' or 'negative' to more nuanced interpretations that capture the complex spectrum of human emotions. This literature review explores significant advancements in aspect-based sentiment analysis, focusing on methodological innovations and model architectures that enhance the understanding and processing of textual data.

Initially, traditional sentiment analysis often relied on binary classifications, as seen in the use of SVMs to determine sentiment polarity. While these models were effective in structured scenarios, they struggled with the unstable and differentiated nature of textual data.

Transitioning from traditional methods, the work of Wang and Liu (n.d.) introduced a hybrid structure that combines syntactic structure analysis with convolutional neural networks. This represents a significant shift towards capturing the subtle emotional nuances in text, paving the way for more complex analyses. Following that, N. U. Pannala and Krishnadeva (2016) also used supervised learning such as SVM and Maximum Entropy (ME) to classify the extracted features. On the other hand, Rybakov and Malafeev (2018) extended the semantic capabilities of sentiment analysis models by refining the construction of word vector spaces. This allowed for a more nuanced weight adjustment and term dynamism, particularly in the analysis of Russian hotel reviews, indicating a move towards more linguistically aware models.

Further enhancing model capabilities, Phan and Ogunbona (2020) introduced a unified framework that integrates aspect extraction and sentiment classification. Utilizing contextual embeddings and syntactic features, this approach significantly improves the detection and analysis of sentiment and aspect terms, showcasing the integration of deep learning techniques with traditional linguistic models. Expanding the application scope, Yang et al. (2019) proposed a Segment-Level Joint Topic-Sentiment Model (STSM). This model analyzes sentiments within segmented parts of sentences, thereby allowing for a granular understanding of sentiment transitions and orientations across different sentence segments, emphasizing the importance of context in sentiment analysis. Additionally, the study by Chong, Selvaretnam, and Soon (2014) demonstrated the practical application of sentiment analysis on social media platforms like Twitter. Here, data pre-processing and sentiment classification techniques were specially adapted to handle the informal and noisy nature of online communication, reflecting the challenges and adaptations required for real-world applications.

There are many challenges encountered by the current implementation of ABSA such as handling expressions that rely on context, sarcasm identification, and language subtleties unique to a certain domain Farhadloo and Rolland (2016). Hence, we are trying to solve those problems through our proposed methods.

3 Methods

3.1 Bag-of-Words with a simple neural network and CNN

First, we did an experiment by using Bag-of-Words (BOW) model along with a simple neural network and Convolutional Neural Network (CNN). BOW manually extracts the word patterns from the text data by tokenize the text to break each text into individual words or tokens. For a simple neural network, we have one hidden layer consisting of 16 neurons and taking 2000 input features. Second, we trained a CNN model with a single Conv1D layer. We limited the length of each review to 600 words and add 0 padding to the end of shorter reviews. We then reshaped each review into a 1D array and fed it into the model. To enable the model have enough space to extract patterns effectively, we set the kernel size to 5. Additionally, we included a MaxPooling1D layer and a dropout layer to regularize the model and prevent over-fitting. Last, in order to improve the accuracy of CNN model and transform the input word space, we added a embedding layer into the CNN model.

3.2 Graph Neutral Network model

For one of our practices, the graph neutral network is used for the sentiment classification and it's shown in Figure1. For each text is represented as a graph, where nodes correspond to word combinations within the text. The edges between these nodes are weighted based on the proximity of the words within the text. Specifically, the weight of an edge is inversely proportional to the distance between words. The formula used to compute the weight between nodes i and j is given by:

$$\text{Weight}_{i,j} = \frac{\text{Number of words} - \text{Word Distance } i \text{ and } j}{\text{Number of words}}$$

This graph structure leverages node embeddings to represent each word combination. Each word is fully connected to every other word, with edge weights calculated as described.

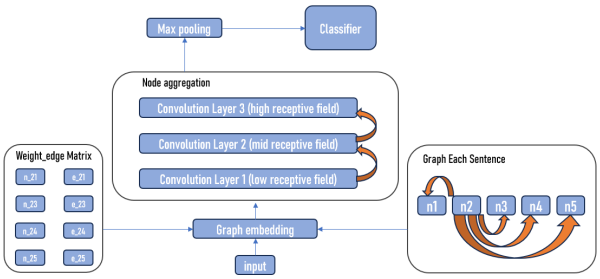

Figure 1: Graph setup

1. **Three Graph Convolution Layers**: There are three layers with three increasing receptive fields from 0.1 to 0.4 and finally to 0.99. The first layer focuses on aggregating information from nodes that are closely related in the text. And second layer extends its aggregation to nodes. The final layer has a receptive field of 0.9999, enabling it to aggregate information across almost the entire graph

   Each layer's output is passed as input to the next layer. Specifically, the nodes aggregated in the first layer are passed to the second layer, and the nodes aggregated in the second layer are passed to the third layer. This hierarchical aggregation ensures that information from different proximity levels is effectively captured and utilized.

2. **Pooling and Output Layer**: Following the three graph convolutional layers, a global max pooling layer is applied to condense the graph's information into a fixed-size representation. The pooled representation is then passed through a dense layer with a sigmoid activation function to produce the final classification output.

$$\text{Output} = \sigma(\text{Dense}(\text{GlobalMaxPooling}(\mathbf{H}^{(3)})))$$

3.3 Lexical Resources Model

This experiment provides an extensive examination that employs the VADER sentiment analysis tool and Linear Regression to perform sentiment analysis on IMDB movie reviews.The process commences with downloading and preparing the IMDB movie reviews dataset. The script then processes movie review files, calculates sentiment scores using VADER, and stores pertinent data in a structured format. The data is subsequently prepared for model training by organising sentiment scores and transforming categorical results into binary format. A Linear Regression model is then trained using the rating and sentiment scores as features and the result as the target variable. The trained model is assessed using test data, and the script reports the model's accuracy in terms of the R-squared value. The entire process is orchestrated seamlessly, ensuring efficient execution. VADER, a lexicon and rule-based tool, is particularly adept at analysing sentiments expressed in social media, assigning scores between -1 (most negative) and +1 (most positive) (Hutto & Gilbert, 2014). Linear Regression, a statistical method in machine learning, predicts a dependent variable based on one or more independent variables. In sentiment analysis, it enables models to learn to associate features with sentiments based on training data, facilitating accurate sentiment prediction in text.

4 Experiments

There will be three different experiments to be conducted for this research paper. First, it will utilize BOW model along with the simple neural network and CNN for predicting the sentiment. Later on, the performance of each model will be measured by its performance on predicting the data through the unseen data which is the test set. Next, we do the experiments for our proposed model. First, we will use the graph neural network model. For the last one, we will use (explain). The results will be explained in details and compared on 4.6.

4.1 IMDB movie review dataset

The dataset to conduct this experiment is obtained from Learning Word Vectors for Sentiment Analysis by A. L. Maas and Potts (2011). It consists of supervised and unsupervised dataset. However, only the supervised dataset will be used for this experiment as we can easily calculate the performance of our proposed model. The supervised dataset consist of 25000 files for training and test respectively. Additionally, it has 12500 of positive and 12500 of negative target value for both of the datasets. As from that data, we can conclude that the data is well-balanced. The positive data is indicated by value 1 and 0 for the negative.

4.2 Experimental setup

To conduct this experiment, we are using Python as the programming language as it has powerful libraries to build the model. Then, we will use the authors' dataset exactly as it is as it has already been split into equal halves for training and testing. However, 20% of the training

dataset will be reserved for validation. This guarantees that throughout training, our model is assessed on untested data, enabling us to assess its performance and make the required corrections for the best outcomes.

4.3 Experiment 1

In this experiment, we treat our sequence text data as 1D images and use CNN for sentiment analysis. We aim to determine if CNN performs well on sequential text data, and compare the results with BoW models with a simple neural network.

First, to establish a baseline for comparison, we repeated the BoW experiments as described in Chollet (2021, pp. 322-325) and trained two BoW models with n-grams to 2 and 3, respectively. There are thousand of keywords extracted such as: "this film","well","me","great", "character" and "bad".

Then we trained a CNN model with a single Conv1D layer as mentioned in method 3.1 and found that it performed poorly in sentiment analysis, achieving only 50% level of accuracy. We identified that this issue might be related to our data. In image processing, similar pixels are close to each other, whereas in our dataset, words are encoded based on their frequency, so the distances between words do not reflect their similarity. As a result, our CNN model might struggle to recognize patterns effectively.

To address this issue, we trained another model by adding an embedding layer to our CNN model. The output from the embedding layer was then fed into the CNN layer. Since the embedding layer does not require a 2D array, we reshaped the input data back into a sequence array.

4.4 Experiment 2

Since we are using GCN, we are able to capture the relationships between the words. In Figure 2, we visualized the extracted words and the relationships with the words that have been converted to numbers. This form has a central hub or focal point in its core that represents important ideas or feelings that are crucial to the analysis. Nodes create concentric rings or clusters around this core hub, with inner rings denoting more central or significant characteristics. These groups, which are arranged around the main topic, are representative of similar features or feelings.

4.5 Experiment 3

We need to pre-process the data into the structured format before feed the data for training in the regression model.

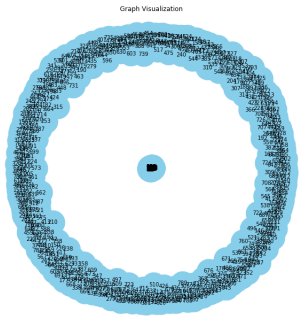

Figure 2: A visualization from GCN

Hence, we tried to classify the sentiment for each data in the train set and give the scores for the positive, negative, neutral and also compound.

Table 1: Example of the score given based on each review

Score	neg	neu	pos	compound
	0.063	0.686	0.251	0.8720
	0.077	0.797	0.125	0.8102
	0.162	0.719	0.119	-0.8575

A vocabulary measure known as the compound score is the total of all positive and negative attitudes adjusted between +1 (positive) and -1 (negative). The sentiment is classed as negative if the compound score is less than 0. If not, it will be advantageous. On the other hand, it is considered neutral if the value is zero. Next, based on the this information, we will build a linear regression on top of it.

4.6 Results

4.6.1 Experiment 1

The results of BoWs using a simple neural network can be seen on Table 2. From Table 2, it achieves the best result by having 2-gram indicated by 90% level of accuracy on test set. Confusion matrix for the 2-gram is illustrated on Figure 3, and the model demonstrates balanced performance in correctly classifying both 0 and 1 instances. However, the approximate equal count of false classifying 0 and 1 indicates potential limitations in the model's ability to distinguish between the two classes effectively even though the accuracy is pretty high.

From Table 3, We can see that the pure CNN model performs poorly, achieving only 50% level of accuracy. After adding the embedding layer and transforming the word space, the model achieves an accuracy of 89.0%. However, it still does not outperform the bag-of-words model with 2-grams.

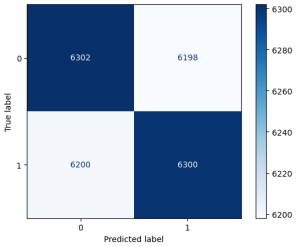

Figure 3: Confusion Matrix of 2-grams on Test Set

Table 2: The Accuracy of Experiment 1 using a simple neural network

Accuracy	BoW1g	BoW2g	BoW3g	BoW4g
Train	0.889	0.901	0.894	0.898
Test	0.889	0.902	0.895	0.899

4.6.2 Experiment 2

We also measured the accuracy of the model on Table 4 by using accuracy tested on the train and test set. However, it looks like the model is under-fitting as it only achieves an accuracy of approximately 50% for both test and training data.

4.6.3 Experiment 3

The high accuracy rate of 88.82% attained by this script underscores the efficacy of combining VADER for sentiment analysis with Linear Regression for prediction. This combination highlights the importance of comprehending semantic context in interpreting sentiment from unstructured data. The validated performance of this potent tool demonstrates its substantial potential for analysing sentiments in movie reviews and other forms of social media text (N. K. Singh, 2020).

5 Discussion

In Experiment 1 mentioned in 4.3, we used an embedding layer to enable the model to learn the relationships between words and transform the input sequence text data simultaneously. However, this is not the only approach. We can also use a pre-trained embedding layer or a pre-trained transformer like BERT for transformation. Additionally, using a residual stack of depthwise-separable 1D convolutional networks for sentiment analysis could also potentially achieve higher accuracy. As Experiment 2, we used GCN as our model. However, it does not meet the expectations since the model is under-fitting. This outcome suggests that the current implementation of GCNs for ABSA might not effectively leverage word relationships

Table 3: The Accuracy of Experiment 1 using CNN

Accuracy	CNN	CNN Embedding
Train	0.507	0.967
Test	0.500	0.890

Table 4: The Accuracy of Experiment 2 using GNN

Accuracy	GNN
Train	0.531
Test	0.513

to improve sentiment classification. In Experiment 3, a model that combined Linear Regression and VADER sentiment analysis achieved an astounding accuracy of 88.82%. This high accuracy highlights the usefulness of combining machine learning methods for sentiment prediction, such Linear Regression, with lexicon-based approaches, like VADER, which are skilled at managing the semantic context of words. The effectiveness of this approach emphasizes how crucial it is for sentiment analysis analysts to comprehend the semantic meaning and context of words. Further research such as implementing Grid Search to find the best parameters is required. Additionally, trying different regression model can also be conducted to get result. Unfortunately, it cannot be done now due to limitations of the computational power and time.

6 Conclusion

This study examined aspect-based sentiment analysis (ABSA) using models such as bag-of-words (BoW), convolutional neural networks (CNNs), graph neural networks (GNNs), and utilizing lexical resources. While BoW models showed high accuracy in sentiment classification, CNNs and GNNs faced challenges with sequential text input and word associations.

Experiment 1 revealed that CNNs require text preprocessing and transformation to capture semantic relationships effectively. Without this, CNNs struggle to identify text patterns. GNNs, particularly GCNs, better capture word interactions than BoW models but still face performance issues. The combination of VADER sentiment analysis and linear regression achieved high accuracy, demonstrating the potential of integrating lexicon-based methods with machine learning for effective sentiment analysis.

In summary, this research adds to the current discussion on ABSA and showing the advantages and disadvantages of various modeling approaches. Researchers may improve sentiment analysis's state-of-the-art and increase its use-

fulness in practical situations by tackling these issues.

References

A. L. Maas, P. T. P. D. H. D. A. Y. N., R. E. Daly, & Potts, C. (2011, June). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies* (pp. 142–150). Portland, Oregon, USA: Association for Computational Linguistics. Retrieved from `http://www.aclweb.org/anthology/P11-1015`

Chollet, F. (2021). *Deep learning with python* (2nd ed.). Shelter Island, NY, USA: Manning Publications Co.

Chong, W. Y., Selvaretnam, B., & Soon, L.-K. (2014). Natural language processing for sentiment analysis: An exploratory analysis on tweets. In *Proceedings of the 2014 4th international conference on artificial intelligence with applications in engineering and technology* (p. 212–217). USA: IEEE Computer Society. Retrieved from `https://doi-org.ezproxy.uow.edu.au/10.1109/ICAIET.2014.43` doi: 10.1109/ICAIET.2014.43

Farhadloo, M., & Rolland, E. (2016, 03). Fundamentals of sentiment analysis and its applications. In (p. 1-24). doi: 10.1007/978-3-319-30319-2_1

Hutto, C., & Gilbert, E. (2014, May). Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media*, *8*(1), 216-225. Retrieved from `https://ojs.aaai.org/index.php/ICWSM/article/view/14550` doi: 10.1609/icwsm.v8i1.14550

M. Al-Smadi, M. A.-A. . Y. J., B. Talafha. (2018). *Using long short-term memory deep neural networks for aspect-based sentiment analysis of arabic reviews.* International Journal of Machine Learning and Cybernetics. doi: https://doi.org/10.1007/s13042-018-0799-4

M. E. Mowlaei, M. S. A., & Keshavarz, H. (2018). A lexicon generation method for aspect-based opinion mining. In *2018 ieee 22nd international conference on intelligent engineering systems (ines)* (p. 000107-000112). doi: 10.1109/INES.2018.8523897

M. Hu, B. L. (2004). *Mining and summarizing customer reviews.* Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. doi: https://dl.acm.org/doi/10.1145/1014052.1014073

N. K. Singh, A. K. S., D. S. Tomar. (2020). Sentiment analysis: a review and comparative analysis over social media. *Journal of Ambient Intelligence and Humanized Computing*.

N. U. Pannala, J. T. K. J. L. R., C. P. Nawarathna, & Krishnadeva, K. (2016). Supervised learning based approach to aspect based sentiment analysis. In *2016 ieee international conference on computer and information technology (cit)* (p. 662-666). doi: 10.1109/CIT.2016.107

Phan, M. H., & Ogunbona, P. O. (2020, July). Modelling context and syntactical features for aspect-based sentiment analysis. In D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 3211–3220). Online: Association for Computational Linguistics. Retrieved from `https://aclanthology.org/2020.acl-main.293` doi: 10.18653/v1/2020.acl-main.293

Rybakov, V., & Malafeev, A. (2018). Aspect-based sentiment analysis of russian hotel reviews. In *Ceur workshop proceedings.*

Wang, B., & Liu, M. (n.d.). Deep learning for aspect-based sentiment analysis. *n.d.*.

W. Medhat, A. H. . H. K. (2014). *Sentiment analysis algorithms and applications: A survey.* Ain Shams Engineering Journal. doi: https://doi.org/10.1016/j.asej.2014.04.011

Yang, Q., Rao, Y., Xie, H., Wang, J., Wang, F. L., & Chan, W. H. (2019). Segment-level joint topic-sentiment model for online review analysis. *IEEE Intelligent Systems*, *34*, 43–50. doi: 10.1109/MIS.2019.2899142