

Graph Convolutional Networks with POS Gate for Aspect-Based Sentiment Analysis

Dahye Kim ¹, YoungJin Kim ¹ and Young-Seob Jeong ^{2,*} 

¹ FS, Inc., Daejeon-si 34126, Korea

² Department of Computer Engineering, Chungbuk National University, Cheongju 28644, Korea

* Correspondence: ysjay@chungbuk.ac.kr

Abstract: We make daily comments on online platforms (e.g., social networks), and such natural language texts often contain sentiment (e.g., positive and negative) for certain aspects (e.g., food and service). If we can automatically extract the aspect-based sentiment from the texts, then it will help many services or products to overcome their limitations of particular aspects. There have been studies of aspect sentiment classification (ASC) that finds sentiment towards particular aspects. Recent studies mostly adopt deep-learning models or graph neural networks as these techniques are capable of capturing linguistic patterns that contributed to performance improvement in various natural language processing tasks. In this paper, for the ASC task, we propose a new hybrid architecture of graph convolutional network (GCN) and recurrent neural network. We design a gate mechanism that jointly models word embeddings and syntactic representation of sentences. By experimental results on five datasets, we show that the proposed model outperforms other recent models and also verify that the gate mechanism contributes to the performance improvement. The overall F1 scores that we achieved is 66.64~76.80%.

Keywords: graph convolutional network; aspect-based sentiment analysis; POS gate; recurrent neural; Word LSTM



Citation: Kim, D.; Kim, Y.; Jeong, Y.-S. Graph Convolutional Networks with POS Gate for Aspect-Based Sentiment Analysis. *Appl. Sci.* **2022**, *12*, 10134. <https://doi.org/10.3390/app121910134>

Academic Editor: Luigi Portinale

Received: 31 August 2022

Accepted: 6 October 2022

Published: 9 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

People often write comments on social network services (SNS) or websites of arbitrary services (e.g., online shopping mall), and it is obvious that it will be significantly helpful for service providers if it is possible to get the opinions or customer satisfaction level from several customer comments. Such comments are mostly written in natural language texts, and an example is shown in Figure 1; there are two aspects (e.g., food and service) with different sentiment (e.g., positive and negative). There have been studies for analyzing the natural language texts containing sentiment and aspects, and such a research field is known as aspect-based sentiment analysis (ABSA). The ABSA contains few tasks [1]: (1) entity-based aspect identification, (2) extraction of linguistic expressions that refer to arbitrary aspects, and (3) aspect sentiment classification (ASC). We aim at the task of ASC that extracts aspect-based sentiment from sentences. A sentence may contain multiple aspects, each of which corresponds to different sentiment, and the aspect-based sentiment (e.g., ‘service’: negative, and ‘food’: positive) will help to examine customer satisfaction; for example, the restaurant owner will investigate what is wrong with the service in the restaurant. Because previous studies might be interpreted in the wrong way if we use different definitions of sentiment, we here follow the definition of Giuseppe D’Aniello et al. [2]; sentiment is a durable emotional disposition that is developed by the user with respect to an aspect.

For the task of ASC whose goal is to identify aspects and sentiment expressed towards each aspect, there have been studies using machine learning models. Kiritchenko et al. [3] used support vector machine (SVM) [4] and achieved 80% accuracy for a dataset of SemEval

2014 Task 4 subtask 2 restaurant domain (REST14). Mubarak et al. [5] used naive Bayes classifier with a feature selection algorithm, and they achieved a 78% of F1 score for the REST14 dataset. Although these studies have shown successful performance, they are limited because they require significant effort in feature engineering.

Deep learning (DL) models are favorable as they find arbitrary features automatically from data [6]. There are a few well-known types of DL models: multi-layered perceptron (MLP) [7], recurrent neural networks (RNN) [8], convolutional neural networks (CNN) [9], and attention mechanism [10]. These different DL models have their own merits, as discussed in [11]; for example, some studies employed CNN to tackle the task of sentiment classification as the CNN is known to be efficient and effective in analyzing local patterns [12,13]. In particular, attention-based RNN architecture was often adopted in many studies of the ASC task as the attention-based architecture has shown its effectiveness in analyzing sequential patterns beneath the word sequences of natural language texts [10,14]. Wang et al. [15] proposed an attention-based RNN model using long short-term memory (LSTM) [16], and achieved 78% accuracy for REST14. Fan et al. [17] proposed bi-directional LSTM (Bi-LSTM) with a fine-grained attention mechanism, that captures the word-level interaction between aspect and context. They achieved 81% accuracy for REST14.

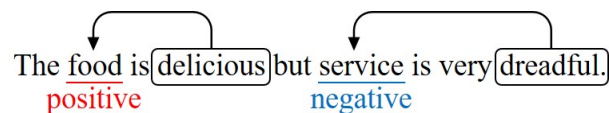


Figure 1. Example of aspect-based sentiment analysis.

Recently, a graph neural network is widely used to incorporate graph-like patterns such as dependency between words for the ASC task. Xing and Tsang [18] proposed a hybrid architecture of LSTM and a graph neural network, and adopted other resources (e.g., DBpedia and Wikipedia) to train their model. They achieved 87% accuracy for REST14. Huang et al. [19] utilized pre-trained Bidirectional Encoder Representations from Transformers (BERT) [20] and graph attention network (GAT) on the dependency tree, and achieved 85% accuracy for REST14. Using context-aware pre-trained language models (e.g., BERT) has drawn much attention as it allows us to achieve better accuracy; however, it requires relatively large computational cost compared to context-free language models (e.g., Word2Vec [21] and Glove [22]), as reported in [23]. There are studies that adopt neither context-aware pre-trained models nor other resources. Zhang et al. [24] proposed building a graph convolutional network (GCN) over the dependency tree of a sentence to exploit syntactical information and word dependencies, and achieved 80% accuracy for the REST14 dataset. Xiao et al. [25] proposed a model of multi-head attention and an attentional-encoding-based GCN of a dependency tree, and achieved 81% accuracy for REST14. Bai et al. [26] proposed a relational graph attention network (RGAT) to incorporate typed syntactic dependencies. They also utilized part-of-speech (POS) tags that are known to convey rich syntactic patterns, and achieved 83% accuracy for REST14. However, they just used embeddings of POS tags, and did not jointly model the POS embeddings and word embeddings.

Word embeddings have often been jointly modeled with syntactic representation (e.g., POS tags) [27,28] because it allows us to capture more sophisticated patterns between semantic and syntactic representation. For example, the word 'good' has a different meaning for different POS tags (e.g., adverb or noun), and it has multiple meaning even for the same POS tag. In this paper, we assume that modeling semantic and syntactic representations jointly will contribute to performance improvement for the ASC task.

We propose a multi-layered GCN-BiLSTM architecture with a gate mechanism that jointly incorporates POS representations and word representations, and proves the effectiveness of the model by experimental results. The proposed architecture consists of GCN and BiLSTM, where the BiLSTM extracts sequential patterns in the given texts and the GCN

analyzes syntactic information. The gate mechanism incorporates the syntactic information and POS tags together, which makes the entire model work more accurately.

2. Method

Figure 2 depicts the framework of our proposed model, namely POS-gated graph convolutional networks (PGGCN), that is composed of two embedding layers (e.g., word embedding and POS embedding) and a hybrid of GCN-BiLSTM layers with the attention mechanism. The word embedding layer and the POS embedding layer convert the given input sequence of words (or POS tags) into real-valued vectors in a different space; thus, we will get two embedding vectors from the embedding layers. Each of the two embedding vectors is delivered to BiLSTM that captures sequential patterns in the embedding vector. The multi-layered GCN and POS-gated attention mechanism exploit the sequential patterns extracted from the BiLSTM models, and generates final representation. Details of each part of the proposed model are provided in the following subsections.

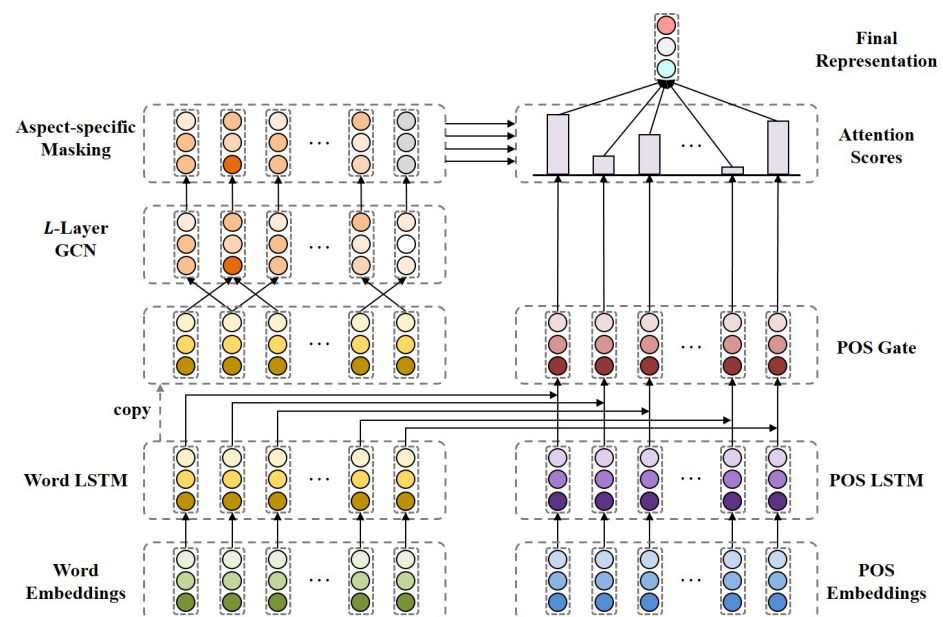


Figure 2. The framework of the proposed model. Word and POS sequences are firstly embedded by embedding layers, and the embedding vectors are processed by two BiLSTM models. Multi-layered GCN and attention mechanism incorporate syntactic patterns and POS tags to generate final representation.

2.1. Embeddings and Bi-LSTM

Given a sentence $\{w_1, w_2, \dots, w_n\}$ where w_i indicates i -th word, we get a sequence of POS tags $\{p_1, p_2, \dots, p_n\}$ using NLTK [29] POS tagger. Aspect is a sub-sequence of the word sequence, and there may exist one or more aspects in a sentence. The words and POS tags are passed to the word embedding layer and POS embedding layer. The embedding layers embed the words and POS tags in vector spaces with word embedding matrix and POS embedding matrix $E^w \in \mathbb{R}^{d_e \times |V|}$, $E^p \in \mathbb{R}^{d_e \times |P|}$, where d_e is the dimension of embedding, $|V|$ denotes the vocabulary size, and $|P|$ indicates the number of unique POS tags. Pre-trained GloVe [22] is used for the word embedding, and the POS embedding is generated using training data with POS tags. The word embeddings and POS embeddings are passed to two different bi-directional LSTM layers (i.e., Word LSTM and POS LSTM), and they yield a sequence of word hidden state vectors $\{h_1^w, h_2^w, \dots, h_n^w\}$ and a sequence of POS hidden state vectors $\{h_1^p, h_2^p, \dots, h_n^p\}$, where $h_i \in \mathbb{R}^k$ is a concatenation of hidden states of forward and backward LSTM at t -th word.

2.2. Gcn and Aspect-Specific Masking

We created an adjacency matrix A between words based on a dependency tree obtained by SpaCy toolkit [30]. The matrix A and word hidden states are fed into the L -layered GCN as shown in the left side of Figure 2. For every l -th layer, i -th word representation h_i^l is updated by incorporating representations of its adjacent words with normalization factor [31] as follows:

$$\tilde{h}_i^l = \sum_{j=1}^n A_{ij} W^l g_j^{l-1} \quad (1)$$

$$h_i^l = \text{Relu}(\tilde{h}_i^l / (d_i + 1) + b^l) \quad (2)$$

where g_j^{l-1} is j -th word representation of previous layer, $d_i = \sum_{j=1}^n A_{ij}$ is a degree of the i -th word in the dependency tree. Note that h_i^l is computed using g_j^{l-1} that is a position-aware word representation [32–34] as follows.

$$g_i^{l-1} = p_i h_i^{l-1} \quad (3)$$

$$p_i = \begin{cases} 1 - \frac{r+1-i}{n} & 1 \leq i < r+1 \\ 0 & r+1 \leq i \leq r+m \\ 1 - \frac{i-r-m}{n} & r+m < i \leq n \end{cases} \quad (4)$$

where $r+1$ is the starting position of an aspect, m is a length of the aspect, and $p_i \in \mathbb{R}$ is the position-aware weight of the i -th word.

At the last GCN layer, we get the final hidden state vectors $H^L = \{h_i^L\}$ where $1 \leq i \leq n$. As shown in the top-left corner in Figure 2, we perform aspect-specific masking that removes non-aspect hidden state vectors; the outputs will be $H_{mask} = \{0, \dots, h_{r+1}^L, \dots, h_{r+m}^L, \dots, 0\}$.

2.3. Part-of-Speech Gate

POS gate is designed to regularize the POS hidden states based on word hidden states. For the i -th word, it takes h_i^w and h_i^p as input, and generates $h_i^{pos} \in \mathbb{R}^m$ as follows.

$$h_i^g = W^g \cdot \tanh(h_i^w + h_i^p) \quad (5)$$

where W^g is a trainable matrix of $\mathbb{R}^{m \times k}$. The gate output is used together with GCN output to generate attention scores as shown in top-right corner of Figure 2.

2.4. Attention-Based Prediction

We compute attention scores for t -th position by incorporating H_{mask} and $H_{gate} = \{h_i^g\}$ through a retrieval-based attention mechanism [24].

$$\beta_t = \sum_{i=1}^n h_t^{g\top} h_i^L = \sum_{i=r+1}^{r+m} h_t^{g\top} h_i^L \quad (6)$$

$$\alpha_t = \frac{\exp(\beta_t)}{\sum_{i=1}^n \exp(\beta_i)} \quad (7)$$

Finally, sentiment $\hat{y}$ is predicted based on h_{att} as formulated below.

$$h_{att} = \sum_{i=1}^n \alpha_i h_i^g \quad (8)$$

$$\hat{y} = \text{softmax}(W h_{att} + b) \quad (9)$$

2.5. Training

We train the model using the cost function C with L_2 -regularization [35] and cross-entropy:

$$C = - \sum_{i=1}^N \sum_{j=1}^S y_i^j \cdot \log(\hat{y}_i^j) + \gamma \|\Theta\|_2 \quad (10)$$

where N is the number of instances, S is the number of classes, $\hat{y}_i^j$ means the output of the model, y_i^j is the ground truth, and γ denotes the L_2 -regularization coefficient.

3. Experiments

3.1. Datasets and Experimental Settings

We conduct experiments on five datasets: restaurant reviews of SemEval 2014, 2015, and 2016 (i.e., REST14, REST15, and REST16) [1,36,37], laptop review of SemEval 2014 (LAPTOP), and Twitter review data (TWITTER) [38]. The Statistics are summarized in Table 1.

The embedding dimension is set to 300, and we borrowed pre-trained GloVe vectors [22] to initialize the word embeddings. The number of GCN layers $L = 2$. We utilized Adam optimizer with the initial learning rate of 10^{-3} . The L_2 regularization coefficient γ is 10^{-5} , and mini-batch size is 32. We use accuracy and macro-averaged F1 score as evaluation metrics, and all results are obtained by averaging 20 independent runs; the F1 score and accuracy are, of course, not the perfect measure, and alternative ways of F-measure [39] may be considered when they are more studied in the future. Our model is compared with others [18,19] that do not employ pre-trained context-aware language models (e.g., BERT) or any other resources (e.g., dataset or knowledge-base).

Table 1. Statistics of the datasets, where m/n indicates the number of training instances m and the number of test instances n .

Dataset	Positive	Neutral	Negative
REST14	2164/728	637/196	807/196
REST15	1178/439	50/35	382/328
REST16	1620/597	88/38	709/190
LAPTOP	994/341	464/169	870/128
TWITTER	1561/173	3127/346	1560/173

3.2. Results

Table 2 shows the overall performance of our model PGGCN and other various models. The PGGCN outperformed other models with all datasets. There are a few interesting points. First, all GCN-based models (e.g., ASGCN, AEGCN, and MGGCN) showed better performance than others; this might imply that the GCN has better ability to grasp underlying syntactic patterns, so it contributes to performance improvement. Second, with the TWITTER dataset, the performance gap between PGGCN and others is smaller compared to other datasets of SemEval. The TWITTER dataset has quite different nature from others; it is collected from social network platform and has much more ‘neutral’ instances, as shown in Table 1. Such many ‘neutral’ instances confuse the models, and may drive the performance gap to be smaller. This is reasonable as ‘neutral’ might not even exist as we are always feeling something [40]. We also observed that all models generally gave better results with greater datasets, except for the TWITTER dataset; this might be related to the many ‘neutral’ instances of the TWITTER dataset.

We performed experiments for the PGGCN without position weight, aspect-specific masking, GCN layers, and POS gate, to check its impact on the performance. The result is summarized in Table 3; it shows that they contributed to performance improvement. In particular, it proves the usefulness of the POS gate, which is the contribution of this paper; modeling word and POS representations jointly helps to get better results on the ASC task.

We also conducted experiments with a varied number of GCN layers, and its accuracies and F1 scores are shown in Figures 3 and 4, respectively. We found that five or more GCN layers did not give any performance gain, and the best number of layers was 2.

Table 2. Averaged accuracies and F1 scores with five datasets (e.g., REST14, REST15, REST16, LAPTOP, and TWITTER), where results with ‡ are borrowed from [24]. The best results for each dataset are in bold. The GCN-based models generally gave better performance than the others, and our proposed PGGCN achieved the best performance.

MODEL	REST14		REST15		REST16		LAPTOP		TWITTER	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
TD-LSTM [41]	78.00	68.43	76.39	58.70	82.16	54.21	71.80	68.46	69.89	66.21
ATAE-LSTM [15]	78.60	67.02	78.48	62.84	83.77	61.71	68.88	63.93	70.14	66.03
MemNet ‡ [33]	79.61	69.64	77.31	58.28	85.44	65.99	70.64	65.17	71.48	69.90
AOA ‡ [42]	79.97	70.42	78.17	57.02	87.50	66.21	72.62	67.52	72.30	70.20
IAN ‡ [43]	79.26	70.09	78.54	52.65	84.74	55.21	72.05	67.38	72.50	70.81
TNet-LF ‡ [32]	80.42	71.03	78.47	59.47	89.07	70.43	74.61	70.14	72.98	71.43
ASGCN ‡ [24]	80.77	72.02	79.89	61.89	88.99	67.48	75.55	71.05	72.15	70.40
AE GCN [25]	81.04	71.32	79.95	60.87	87.39	68.22	75.91	71.63	73.16	71.82
MG GCN [44]	81.16	71.73	80.19	64.62	88.96	69.48	75.80	71.75	73.41	71.89
PGGCN	83.84	76.80	82.47	66.64	90.42	74.49	77.74	74.56	74.57	72.01

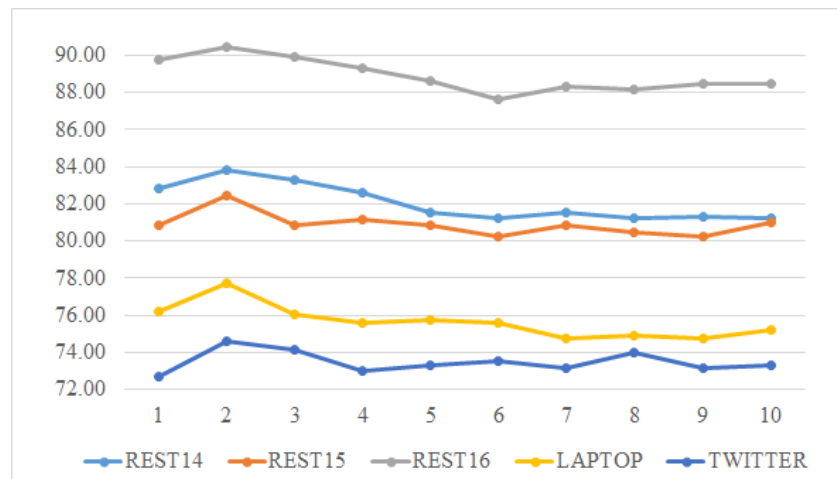


Figure 3. Accuracies with varied number of GCN layers and different datasets, where horizontal axis indicates the number of GCN layers, and the vertical axis represents accuracy.

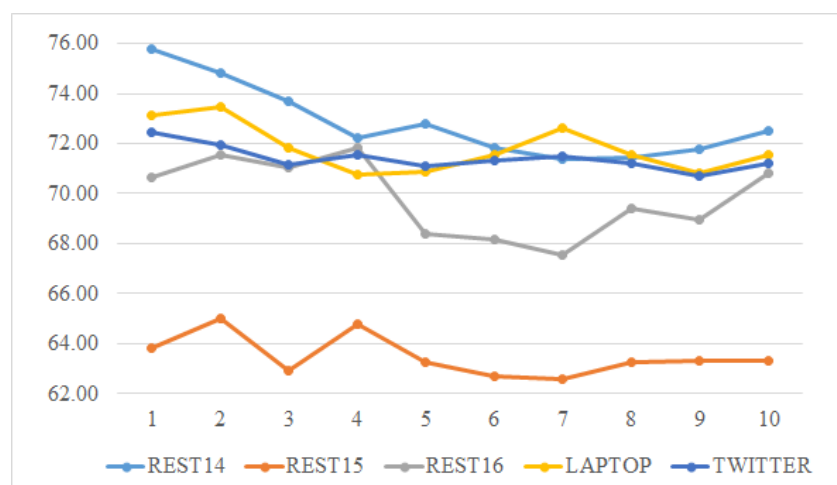


Figure 4. F1 scores with varied number of GCN layers and different datasets, where horizontal axis indicates the number of GCN layers and the vertical axis represents F1 score.

Table 3. Averaged accuracies and F1 scores of PGGCN without some components, where ‘w/o POS gate’, ‘w/o position’, ‘w/o mask’, and ‘w/o GCN’ indicate PGGCN without POS gate, without position weights, without aspect-specific masking, and without GCN layers, respectively. The PGGCN without any component gave the worst results, which implies that all components including the POS gate are important.

MODEL	REST14		REST15		REST16		LAPTOP		TWITTER	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
w/o POS gate	83.39	76.37	80.81	65.14	89.45	73.79	75.86	72.26	73.12	71.62
w/o position	82.68	74.82	80.81	64.30	89.45	71.11	75.39	71.07	73.84	72.29
w/o mask	79.29	69.85	78.04	63.52	87.50	64.99	72.41	67.88	72.40	71.13
w/o GCN	81.25	72.47	81.00	63.34	87.82	69.67	74.61	70.69	72.40	71.30

4. Conclusions

As a solution for the ASC task, we propose a hybrid architecture of GCN-BiLSTM, namely PGGCN, with a gate mechanism that jointly exploits word representations and POS representations. We compared PGGCN with other recent models by experiments with five datasets, and showed that PGGCN outperformed the others. Amongst the many models, we found that GCN-based models outperformed others, which indicates that syntactic patterns discovered by GCN has an impact on the performance. We also observed that the models generally gave better results with greater datasets, but the models gave poor results when the dataset has many ‘neutral’ instances. The experimental results of PGGCN without the POS gate showed the effectiveness of the POS gate. We did not utilize any pre-trained language models [20,45] that have shown promising results recently, so we will investigate a way of combining our model with pre-trained context-aware language models as future work.

Author Contributions: Conceptualization, D.K. and Y.-S.J.; methodology, D.K. and Y.-S.J.; validation, D.K. and Y.-S.J.; resources, Y.K.; writing—original draft preparation, D.K.; writing—review and editing, D.K. and Y.-S.J.; funding acquisition, Y.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2020R111A3053015).

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pontiki, M.; Galanis, D.; Papageorgiou, H.; Androutsopoulos, I.; Manandhar, S.; Al-Smadi, M.; Al-Ayyoub, M.; Zhao, Y.; Qin, B.; Clercq, O.D.; et al. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In Proceedings of the 10th International Workshop on Semantic Evaluation, San Diego, CA, USA, 16–17 June 2016; pp. 19–30.
2. D’Aniello, G.; Gaeta, M.; Rocca, I.L. KnowMIS-ABSA: An overview and a reference model for applications of sentiment analysis and aspect-based sentiment analysis. *Artif. Intell. Rev.* **2022**, *55*, 5543–5574. [\[CrossRef\]](#)
3. Kiritchenko, S.; Zhu, X.; Cherry, C.; Mohammad, S. NRC-Canada-2014: Detecting Aspects and Sentiment in Customer Reviews. In Proceedings of the 8th International Workshop on Semantic Evaluation, Dublin, Ireland, 23–24 August 2014; pp. 437–442.
4. Burges, C.J.C. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Min. Knowl. Discov.* **1998**, *2*, 121–167. [\[CrossRef\]](#)
5. Mubarak, M.S.; Adiwijaya; Aldhi, M.D. Aspect-based sentiment analysis to review products using Naïve Bayes. *AIP Conf. Proc.* **2017**, *1867*, 020060.
6. Sarker, I.H. Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN Comput. Sci.* **2021**, *2*, 1–20. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Haykin, S.S. *Neural Networks: A Comprehensive Foundation*; Prentice Hall: Hoboken, NJ, USA, 1999.
8. Medsker, L.; Jain, L.C. *Recurrent Neural Networks: Design and Applications*; CRC Press: Boca Raton, FL, USA, 1999.
9. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-Based Learning Applied to Document Recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [\[CrossRef\]](#)

10. Luong, T.; Pham, H.; Manning, C.D. Effective Approaches to Attention-based Neural Machine Translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 1412–1421.
11. Shrestha, A.; Mahmood, A. Review of Deep Learning Algorithms and Architectures. *IEEE Access* **2019**, *7*, 53040–53065. [[CrossRef](#)]
12. Sitaula, C.; Basnet, A.; Mainali, A.; Shahi, T.B. Deep Learning-Based Methods for Sentiment Analysis on Nepali COVID-19-Related Tweets. *Comput. Intell. Neurosci.* **2021**, *2021*, 1–11. [[CrossRef](#)] [[PubMed](#)]
13. Kim, H.; Jeong, Y.S. Sentiment Classification Using Convolutional Neural Networks. *Appl. Sci.* **2019**, *9*, 2347. [[CrossRef](#)]
14. Galassi, A.; Lippi, M.; Torrioni, P. Attention in Natural Language Processing. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 4291–4308. [[CrossRef](#)] [[PubMed](#)]
15. Wang, Y.; Huang, M.; Zhu, X.; Zhao, L. Attention-based LSTM for Aspect-level Sentiment Classification. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 606–615.
16. Hochreiter, S.; Schmidhuber, J. Long Short-term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
17. Fan, F.; Feng, Y.; Zhao, D. Multi-grained Attention Network for Aspect-Level Sentiment Classification. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 3433–3442.
18. Xing, B.; Tsang, I.W. Understand me, if you refer to Aspect Knowledge: Knowledge-aware Gated Recurrent Memory Network. *IEEE Trans. Emerg. Top. Comput. Intell.* **2022**, *6*, 1092–1102. [[CrossRef](#)]
19. Huang, L.; Sun, X.; Li, S.; Zhang, L.; Wang, H. Syntax-Aware Graph Attention Network for Aspect-Level Sentiment Classification. In Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain, 8–13 December 2020; pp. 799–810.
20. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
21. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. *arXiv* **2013**, arXiv:1301.3781.
22. Pennington, J.; Socher, R.; Manning, C. GloVe: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
23. Rae, J.W.; Borgeaud, S.; Cai, T.; Millican, K.; Hoffmann, J.; Song, F.; Aslanides, J.; Henderson, S.; Ring, R.; Young, S.; et al. Scaling Language Models: Methods, Analysis & Insights from Training Gopher. *arXiv* **2021**, arXiv:2112.11446.
24. Zhang, C.; Li, Q.; Song, D. Aspect-based Sentiment Classification with Aspect-specific Graph Convolutional Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, Hong Kong, China, 3–7 November 2019; pp. 4568–4578.
25. Xiao, L.; Hu, X.; Hu, Y.; Xue, Y.; Gu, D.; Chen, B.; Zhang, T. Targeted Sentiment Classification Based on Attentional Encoding and Graph Convolutional Networks. *Appl. Sci.* **2020**, *10*, 957. [[CrossRef](#)]
26. Bai, X.; Liu, P.; Zhang, Y. Exploiting Typed Syntactic Dependencies for Targeted Sentiment Classification Using Graph Attention Neural Network. *arXiv* **2020**, arXiv:2002.09685.
27. Liu, Q.; Ling, Z.H.; Jiang, H.; Hu, Y. Part-of-Speech Relevance Weights for Learning Word Embeddings. *arXiv* **2016**, arXiv:1603.07695.
28. Smith, A.; de Lhoneux, M.; Stymne, S.; Nivre, J. An Investigation of the Interactions Between Pre-Trained Word Embeddings, Character Models and POS Tags in Dependency Parsing. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 2711–2720.
29. Bird, S.; Loper, E. NLTK: The Natural Language Toolkit. In Proceedings of the ACL Interactive Poster and Demonstration Sessions, Barcelona, Spain, 21–26 July 2004; pp. 214–217.
30. Honnibal, M.; Montani, I. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. *To Appear* **2017**, *7*, 411–420.
31. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv* **2016**, arXiv:1609.02907.
32. Li, X.; Bing, L.; Lam, W.; Shi, B. Transformation Networks for Target-Oriented Sentiment Classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; Volume 1, pp. 946–956.
33. Tang, D.; Qin, B.; Liu, T. Aspect Level Sentiment Classification with Deep Memory Network. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 214–224.
34. Chen, P.; Sun, Z.; Bing, L.; Yang, W. Recurrent Attention Network on Memory for Aspect Sentiment Analysis. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017; pp. 452–461.
35. Ng, A.Y. Feature selection, L1 vs. L2 regularization, and rotational invariance. In Proceedings of the Twenty-First International Conference on Machine Learning, Banff, AB, Canada, 4–8 July 2004.
36. Pontiki, M.; Galanis, D.; Pavlopoulos, J.; Papageorgiou, H.; Androutsopoulos, I.; Manandhar, S. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In Proceedings of the 8th International Workshop on Semantic Evaluation, Dublin, Ireland, 23–24 August 2014; pp. 27–35.
37. Pontiki, M.; Galanis, D.; Papageorgiou, H.; Manandhar, S.; Androutsopoulos, I. SemEval-2015 Task 12: Aspect Based Sentiment Analysis. In Proceedings of the 9th International Workshop on Semantic Evaluation, Denver, CO, USA, 4–5 June 2015; pp. 486–495.

38. Dong, L.; Wei, F.; Tan, C.; Tang, D.; Zhou, M.; Xu, K. Adaptive Recursive Neural Network for Target-dependent Twitter Sentiment Classification. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 22–27 June, 2014; Volume 2, pp. 49–54.
39. Hand, D.; Christen, P. A note on using the F-measure for evaluating record linkage algorithms. *Stat. Comput.* **2018**, *28*, 539–547. [[CrossRef](#)]
40. Gasper, K.; Spencer, L.A.; Hu, D. Does neutral affect exist? How challenging three beliefs about neutral affect can advance affective research. *Front. Psychol.* **2019**, *10*, 2476. [[CrossRef](#)] [[PubMed](#)]
41. Tang, D.; Qin, B.; Feng, X.; Liu, T. Effective LSTMs for Target-Dependent Sentiment Classification. In Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers, Osaka, Japan, 11–16 December 2016; pp. 3298–3307.
42. Huang, B.; Ou, Y.; Carley, K.M. Aspect Level Sentiment Classification with Attention-over-Attention Neural Networks. In Proceedings of the Social, Cultural, and Behavioral Modeling, Washington, DC, USA, 10–13 July 2018; pp. 197–206.
43. Ma, D.; Li, S.; Zhang, X.; Wang, H. Interactive Attention Networks for Aspect-Level Sentiment Classification. In Proceedings of the 26th International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; pp. 4068–4074.
44. Xiao, L.; Hu, X.; Chen, Y.; Xue, Y.; Chen, B.; Gu, D.; Tang, B. Multi-head self-attention based gated graph convolutional networks for aspect-based sentiment classification. *Multimed. Tools Appl.* **2020**, *81*, 19051–19070. [[CrossRef](#)]
45. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Unsupervised Multitask Learners. *arXiv* **2020**, arXiv:2005.14165.