A Position Aware Decay Weighted Network for Aspect Based Sentiment Analysis

Avinash Madasu and Vijjini Anvesh Rao[✉]

Samsung R&D Institute, Bangalore, India
{m.avinash,a.vijjini}@samsung.com

Abstract. Aspect Based Sentiment Analysis (ABSA) is the task of identifying sentiment polarity of a text given another text segment or aspect. In ABSA, a text can have multiple sentiments depending upon each aspect. Aspect Term Sentiment Analysis (ATSA) is a subtask of ABSA, in which aspect terms are contained within the given sentence. Most of the existing approaches proposed for ATSA, incorporate aspect information through a different subnetwork thereby overlooking the advantage of aspect terms' presence within the sentence. In this paper, we propose a model that leverages the positional information of the aspect. The proposed model introduces a decay mechanism based on position. This decay function mandates the contribution of input words for ABSA. The contribution of a word declines as farther it is positioned from the aspect terms in the sentence. The performance is measured on two standard datasets from SemEval 2014 Task 4. In comparison with recent architectures, the effectiveness of the proposed model is demonstrated.

Keywords: Aspect Based Sentiment Analysis · Attention · Sentiment Analysis · Text classification

1 Introduction

Text Classification deals with the branch of Natural Language Processing (NLP) that involves classifying a text snippet into two or more predefined categories. Sentiment Analysis (SA) addresses the problem of text classification in the setting where these predefined categories are sentiments like positive or negative [7]. Aspect Based Sentiment Analysis (ABSA) is proposed to perform sentiment analysis at an aspect level [2]. There are four sub-tasks in ABSA namely Aspect Term Extraction (ATE), Aspect Term Sentiment Analysis (ATSA), Aspect Category Detection (ACD), Aspect Category Sentiment Analysis (ACSA). In the first sub-task (ATE), the goal is to identify all the aspect terms for a given sentence. Aspect Term Sentiment Analysis (ATSA) is a classification problem where given an aspect and a sentence, the sentiment has to classified into one of the predefined polarities. In the ATSA task, the aspect is present within the sentence but

can be a single word or a phrase. In this paper, we address the problem of ATSA. Given a set of aspect categories and a set of sentences, the problem of ACD is to classify the aspect into one of those categories. ACSA can be considered similar to ATSA, but the aspect term may not be present in the sentence. It is much harder to find sentiments at an aspect level compared to the overall sentence level because the same sentence might have different sentiment polarities for different aspects. For example consider the sentence, *"The taste of food is good but the service is poor"*. If the aspect term is *food*, the sentiment will be *positive*, whereas if the aspect term is *service*, sentiment will be *negative*. Therefore, the crucial challenge of ATSA is modelling the relationship between aspect terms and its context in the sentence. Traditional methods involve feature engineering trained with machine learning classifiers like Support Vector Machines (SVM) [4]. However, these methods do not take into account the sequential information and require a considerable struggle to define the best set of features. With the advent of deep learning, neural networks are being used for the task of ABSA. For ATSA, LSTM coupled with attention mechanism [1] have been widely used to focus on words relevant to certain aspect. Target-Dependent Long Short-Term Memory (TD-LSTM) uses two LSTM networks to model left and right context words surrounding the aspect term [12]. The outputs from last hidden states of LSTM are concatenated to find the sentiment polarity. Attention Based LSTM (ATAE-LSTM) uses attention on the top of LSTM to concentrate on different parts of a sentence when different aspects are taken as input [15]. Aspect Fusion LSTM (AF-LSTM) [13] uses associative relationship between words and aspect to perform ATSA. Gated Convolution Neural Network (GCAE) [17] employs a gated mechanism to learn aspect information and to incorporate it into sentence representations.

However, these models do not utilize the advantage of the presence of aspect term in the sentence. They either employ an attention mechanism with complex architecture to learn relevant information or train two different architectures for learning sentence and aspect representations. In this paper, we propose a model that utilizes the positional information of the aspect in the sentence. We propose a parameter-less decay function based learning that leverages the importance of words closer to the aspect. Hence, evading the need for a separate architecture for integrating aspect information into the sentence. The proposed model is relatively simple and achieves improved performance compared to models that do not use position information. We experiment with the proposed model on two datasets, restaurant and laptop from SemEval 2014.

2   Related Work

2.1   Aspect Term Sentiment Analysis

Early works of ATSA, employ lexicon based feature selection techniques like Parts of Speech Tagging (POS), unigram features and bigram features [4]. However, these methods do not consider aspect terms and perform sentiment analysis on the given sentence.

Phrase Recursive Neural Network for Aspect based Sentiment Analysis (PhraseRNN) [6] was proposed based on Recursive Neural Tensor Network [10] primarily used for semantic compositionality. PhraseRNN uses dependency and constituency parse trees to obtain aspect representation. An end-to-end neural network model was introduced for jointly identifying aspect and polarity [9]. This model is trained to jointly optimize the loss of aspect and the polarity. In the final layer, the model outputs one of the sentiment polarities along with the aspect. [14] introduced Aspect Fusion LSTM (AF-LSTM) for performing ATSA.

3   Model

In this section, we propose the model Position Based Decay Weighted Network (PDN). The model architecture is shown in Fig. 2. The input to the model is a sentence S and an Aspect A contained within it. Let n represent the maximum sentence length considered.

3.1   Word Representation

Let V be the vocabulary size considered and $X \in \mathbb{R}^{V \times d_w}$ represent the embedding matrix[1], where for each word X_i is a d_w dimensional word vector. Words contained in the embedding matrix are initialized to their corresponding vectors whereas words not contained are initialized to 0's. $I \in \mathbb{R}^{n \times d_w}$ denotes the pretrained embedding representation of a sentence where n is the maximum sentence length.

3.2   Position Encoding

In the ATSA task, aspect A is contained in the sentence S. A can be a word or a phrase. Let k_s denote the starting index and k_e denote the ending index of the aspect term(s) in the sentence. Let i be the index of a word in the sentence. The position encoding of words with respect to aspect are represented using the formula

$$p(i) = \begin{cases} k_s - i + 1, & k_s > i \\ 1, & i \in k_s, k_{s+1}, .., k_{e-1}, k_e \\ i - k_e + 1, & i > k_e \end{cases} \qquad (1)$$

The position encodings for the sentence "granted the space is smaller than most it is the best service" where "space" is the aspect is shown in Fig. 2. This number reflects the relative distance of a word from the closest aspect word. The position embeddings from the position encodings are randomly initialized and updated during training. Hence, $P \in \mathbb{R}^{n \times d_p}$ is the position embedding representations of the sentence. d_p denotes the number of dimensions in the position embedding.

---

[1] https://nlp.stanford.edu/data/glove.840B.300d.zip.

3.3   Architecture

As shown in Fig. 2, PDN comprises of two sub-networks: Position Aware Attention Network (PDN) and Decay Weighting Network (DWN).

Position Aware Attention Network (PAN). An LSTM layer is trained on I to produce hidden state representation $h_t \in \mathbb{R}^{d_h}$ for each time step $t \in \{1, n\}$ where d_h is the number of units in the LSTM. The LSTM outputs contain sentence level information and Position embedding contain aspect level information. An attention subnetwork is applied on all h and P to get a scalar score α indicating sentiment weightage of the particular time step to the overall sentiment. However, prior to concatenation, the position embeddings and the LSTM outputs may have been output from disparate activations leading to different distribution. Training on such values may bias the network towards one of the representations. Therefore, we apply a fully connected layer separately but with same activation function Scaled Exponential Linear Unit (SELU) [5] upon them. Two fully connected layers follow this representation. Following are the equations that produce α from LSTM outputs h and position embeddings P.

$$P'_t = selu(W_p \cdot P_t + b_p) \tag{2}$$

$$h'_t = selu(W_h \cdot h_t + b_h) \tag{3}$$

$$H_t = relu(W_a \cdot [h'_t P'_t] + b_a) \tag{4}$$

$$e_t = \tanh(\mathbf{v}^\mathsf{T} \cdot H_t) \tag{5}$$

$$\alpha_t = \frac{\exp(e_t)}{\sum_{i=1}^{n} \exp(e_i)} \tag{6}$$

Decay Weighting Network (DWN). In current and following section, we introduce decay functions. The decay function for scalar position encoding $p(i)$ is represented as the scalar $d(p(i))$. These functions are continuously decreasing in the range $[0, \infty)$. The outputs from the LSTM at every time step are scaled by the decay function's output.

$$Z_t = h_t \cdot d(p(t)) \,\forall\, t \in \{1, n\} \tag{7}$$

A weighted sum O is calculated on the outputs of Decay Weighted network using the attention weights from PAN.

$$O = \alpha \cdot Z \tag{8}$$

A fully connected layer is applied on O which provides an intermediate representation Q. A softmax layer is fully connected to this layer to provide final probabilities.
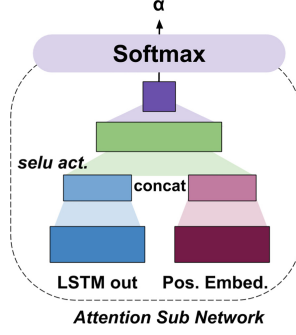
Fig. 1. Attention sub network

Table 1. Accuracy Scores of all models. Performances of baselines are cited from [13]

Model	Restaurant	Laptop
Majority	65.00	53.45
NBOW	67.49	58.62
LSTM	67.94	61.75
TD-LSTM	69.73	62.38
AT-LSTM	74.37	65.83
ATAE-LSTM	70.71	60.34
DCNN	75.18	64.67
AF-LSTM	75.44	68.81
GCAE	76.07	67.27
Tangent-PDN	78.12	68.82
Inverse-PDN	**78.9**	**70.69**
Expo-PDN	78.48	69.43

It is paramount to note that the DWN does not contain any parameters and only uses a decay function and multiplication operations. The decay function provides us with a facility to automatically weight representations closer to aspect as higher and far away as lower, as long as the function hyperparameter is tuned fittingly. Lesser parameters makes the network efficient and easy to train.

Decay Functions. We performed experiments with the following decay functions.

Inverse Decay
Inverse decay is represented as:
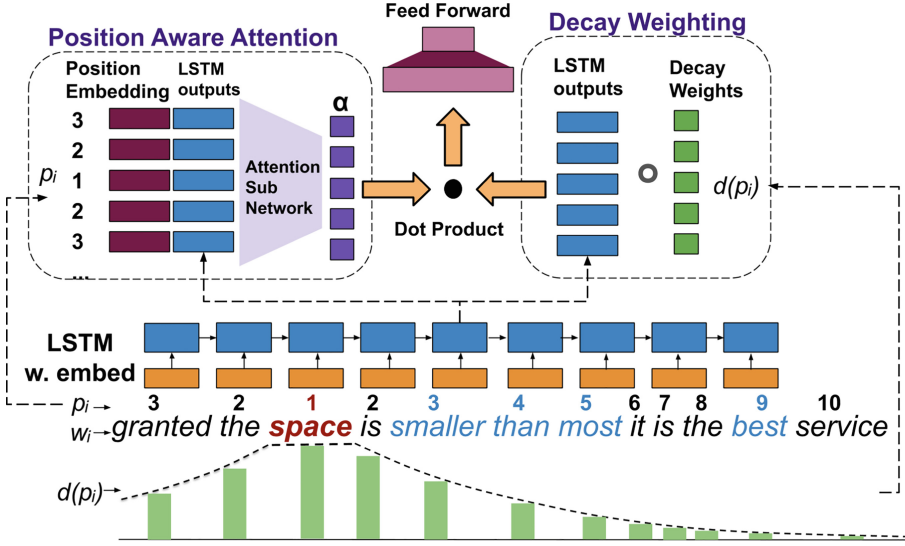
$$d(x) = \frac{\lambda}{x} \tag{9}$$

Fig. 2. PDN Architecture, in the shown example, "space" is the aspect. Refer to Fig. 1 for the Attention Sub Network.

Exponential Decay

Exponential decay is represented as:

$$d(x) = e^{-\lambda * x} \tag{10}$$

Tangent Decay

Tangent decay is represented as:

$$d(x) = 1 - tanh(\lambda * x) \tag{11}$$

λ is the hyper-parameter in all the cases.[2]

4   Experiments

4.1   Datasets

We performed experiments on two datasets, Restaurant and Laptop from SemEval 2014 Task 4 [8]. Each data point is a triplet of sentence, aspect and sentiment label. The statistics of the datasets are shown in the Table 2. As most existing works reported results on three sentiment labels *positive,negative,neutral* we performed experiments by removing *conflict* label as well.

---

[2] In our experiments we took $\lambda = 0.45$ for Tangent-PDN, 1.1333 for Inverse-PDN and 0.3 for Expo-PDN.

4.2   Compared Methods

We compare proposed model to the following baselines:

Neural Bag-of-Words (NBOW). NBOW is the sum of word embeddings in the sentence [13].

LSTM. Long Short Term Memory (LSTM) is an important baseline in NLP. For this baseline, aspect information is not used and sentiment analysis is performed on the sentence alone. [13].

TD-LSTM. In TD-LSTM, two separate LSTM layers for modelling the preceding and following contexts of the aspect is done for aspect sentiment analysis [12].

AT-LSTM. In Attention based LSTM (AT-LSTM), aspect embedding is used as the context for attention layer, applied on the sentence [15].

ATAE-LSTM. In this model, aspect embedding is concatenated with input sentence embedding. LSTM is applied on the top of concatenated input [15].

Table 2. Statistics of the datasets

Dataset	Positive		Negative		Neutral	
	Train	Test	Train	Test	Train	Test
Restaurant	2164	728	805	196	633	196
Laptop	987	341	866	128	460	169

DE-CNN. Double Embeddings Convolution Neural Network (DE-CNN) achieved state of the art results on aspect extraction. We compare proposed model with DE-CNN to see how well it performs against DE-CNN. We used aspect embedding instead of domain embedding in the input layer and replaced the final CRF layer with MaxPooling Layer. Results are reported using author's code[3] [16].

AF-LSTM. AF-LSTM incorporates aspect information for learning attention on the sentence using associative relationships between words and aspect [13].

GCAE. GCAE adopts gated convolution layer for learning aspect representation which is integrated into sentence representation through another gated

---

[3] https://github.com/howardhsu/DE-CNN.

convolution layer. This model reported results for four sentiment labels. We ran the experiment using author's code[4] and reported results for three sentiment labels [17].

4.3   Implementation

Every word in the input sentence is converted to a 300 dimensional vector using pretrained word embeddings. The dimension of positional embedding is set to 25 which is initialized randomly and updated during training. The hidden units of LSTM are set to 100. The number of hidden units in the layer fully connected to LSTM is 50 and the layer fully connected to positional embedding layer is 50. The number of hidden units in the penultimate fully connected layer is set to 64. We apply a dropout [11] with a probability 0.5 on this layer. A batch size 20 is considered and the model is trained for 30 epochs. Adam [3] is used as the optimizer with an initial learning rate 0.001.

5   Results and Discussion

The Results are presented in Table 1. The Baselines Majority, NBOW and LSTM do not use aspect information for the task at all. Proposed models significantly outperform them.

5.1   The Role of Aspect Position

The proposed model outperforms other recent and popular architectures as well, these architectures use a separate architecture which takes the aspect input distinctly from the sentence input. In doing so they loose the positional information of the aspect within the sentence. We hypothesize that this information is valuable for ATSA and our results reflect the same. Additionally since proposed architecture does not take any additional aspect inputs apart from position, we get a fairer comparison on the benefits of providing aspect positional information over the aspect words themselves.

5.2   The Role of Decay Functions

Furthermore, while avoiding learning separate architectures for weightages, decay functions act as good approximates. These functions rely on constants alone and lack any parameters thereby expressing their efficiency. The reason these functions work is because they consider an assumption intrinsic to the nature of most natural languages. It is that description words or aspect modifier words come close to the aspect or the entity they describe. For example in Fig. 2, we see the sentence from the Restaurant dataset, "granted the space is smaller than most, it is the best service you can...". The proposed model is able to handle this example which has distinct sentiments for the aspects "space" and "service" due to their proximity from "smaller" and "best" respectively.

---

[4] https://github.com/wxue004cs/GCAE.

6   Conclusion

In this paper, we propose a novel model for Aspect Based Sentiment Analysis relying on relative positions on words with respect to aspect terms. This relative position information is realized in the proposed model through parameter-less decay functions. These decay functions weight words according to their distance from aspect terms by only relying on constants proving their effectiveness. Furthermore, our results and comparisons with other recent architectures, which do not use positional information of aspect terms demonstrate the strength of the decay idea in proposed model.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
2. Hu, M., Liu, B.: Mining opinion features in customer reviews. In: AAAI, vol. 4, pp. 755–760 (2004)
3. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
4. 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 (SemEval 2014), pp. 437–442 (2014)
5. Klambauer, G., Unterthiner, T., Mayr, A., Hochreiter, S.: Self-normalizing neural networks. In: Advances in Neural Information Processing Systems, pp. 971–980 (2017)
6. Nguyen, T.H., Shirai, K.: PhraseRNN: phrase recursive neural network for aspect-based sentiment analysis. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 2509–2514. Association for Computational Linguistics, Lisbon, September 2015. https://doi.org/10.18653/v1/D15-1298, https://www.aclweb.org/anthology/D15-1298
7. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing, vol. 10, pp. 79–86. Association for Computational Linguistics (2002)
8. 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 (SemEval 2014), pp. 27–35. Association for Computational Linguistics, Dublin, August 2014. https://doi.org/10.3115/v1/S14-2004, https://www.aclweb.org/anthology/S14-2004
9. Schmitt, M., Steinheber, S., Schreiber, K., Roth, B.: Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 1109–1114. Association for Computational Linguistics, Brussels, October–November 2018. https://www.aclweb.org/anthology/D18-1139
10. Socher, R., et al.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1631–1642. Association for Computational Linguistics, Seattle, October 2013. https://www.aclweb.org/anthology/D13-1170

11. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
12. Tang, D., Qin, B., Feng, X., Liu, T.: Effective LSTMs for target-dependent sentiment classification. arXiv preprint arXiv:1512.01100 (2015)
13. Tay, Y., Tuan, L.A., Hui, S.C.: Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
14. Tay, Y., Tuan, L.A., Hui, S.C.: Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis (2018)
15. Wang, Y., Huang, M., Zhao, L., et al.: Attention-based LSTM for aspect-level sentiment classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 606–615 (2016)
16. Xu, H., Liu, B., Shu, L., Yu, P.S.: Double embeddings and CNN-based sequence labeling for aspect extraction. arXiv preprint arXiv:1805.04601 (2018)
17. Xue, W., Li, T.: Aspect based sentiment analysis with gated convolutional networks. arXiv preprint arXiv:1805.07043 (2018)