24th European Conference on Artificial Intelligence - ECAI 2020
Santiago de Compostela, Spain

A Variant of Recurrent Entity Networks for Targeted Aspect-Based Sentiment Analysis

Zhihao Ye [1] and **Zhiyong Li** [2,*]

Abstract. Deep neural network models have achieved promising results on targeted aspect-based sentiment analysis. However, previous models did not effectively match long-distance fine-grained sentiment polarity with the associated target and aspects, and the interdependence among the specific target, corresponding aspects, and the context is always ignored. This work proposes a novel recurrent entity memory network that employs word-level information and sentence-level hidden memory to entity state tracking. In addition, the entity state is utilized to fine-tune the target embedding and aspect embedding. The experimental results showed that the proposed model outperformed previous models.

1 Introduction

Targeted aspect-based sentiment analysis (TABSA) [19] is an important task in natural language processing. Different from traditional coarse-grained sentiment analysis, TABSA identifies fine-grained opinion polarity towards a specific aspect associated with a given target. One example is shown in Figure 1. The sentiment polarity is "Positive" towards the aspect "SAFETY" and the target "location1", "Negative" towards the aspect "PRICE" and the target "location1" and "Negative" polarity towards the aspect "TRANSIT-LOCATION" and the target "location2".

Many different models, designed for (T)ABSA, have been proposed and have achieved promising results. The traditional methods [24, 9] rely heavily on feature engineering and the results are not satisfactory. Recently, with the development of deep learning, an increasing number of researchers have tried to apply different deep learning models [2, 31, 15, 25, 13, 10] to (T)ABSA. One study [22] used long-short term memory (LSTM) to automatically learn the aspect-specific word and sentence representations, and an attention-based sentiment LSTM [14] was proposed to tackle the challenges of both aspect-based sentiment analysis and targeted sentiment analysis by incorporating external knowledge. More recently, the researchers applied memory network [27] and its variants [23, 2] of it to the (T)ABSA task. In particular, the recurrent entity networks [5] model was improved on [12] and a delayed memory update mechanism was proposed that can decouple the duty of capturing transitions of activations between steps from the task of the entity state tracking and monitor the activation of the update gate. Their model used fine-grained word levels as input (i.e., unlike the coarse-grained sentence-level input that the original recurrent entity networks applied to reading comprehension tasks, where the input is word after word in the TABSA task), and mainly used word-level information to track entity states. Their model successfully adapts to the TABSA task and achieves a state-of-the-art effect on the sentihood [19] dataset.

location1 is your best bet for secure although expensive and **location2** is too far

Target	Aspect	Sentiment
location1	SAFETY	Positive
location1	PRICE	Negative
location2	TRANSIT	Negative

Figure 1. Example of TABSA task. Opinions on the aspects SAFETY and PRICE are expressed for entity "location1" but not entity "location2"

However, the model [12] still has some disadvantages. At each recurrent step, information exchange is conducted between consecutive words in the sentence and the entity state is tracked only by word-level information, which leads to a relatively weaker power in capturing long-range dependencies in a TABSA task. Moreover, the models do not consider sentence-level hidden memory, which can help to track the entity state. Finally, many prior works typically utilize context-independent or randomly initialized vectors to represent targets and aspects, which loses the semantic information and ignores the interdependence among a specific target, corresponding aspects, and the context. As shown in the example in Figure 1, the opinion "Positive" forward the aspect "SAFETY" is expressed for the target "location1" will be change if "location1" and "location2" are exchanged. In other words, the opinions of the given sentence are generally composed of words highly correlative to the targets and aspects as the targets and aspects themselves have no expression of sentiment.

After observing TABSA's task-based data, it can be seen that many sentences are long, and the corresponding sentiment polarity of a certain target or aspect can only be reflected by a long distance. For example, in Figure 1, the sentence starts with "location1", but the negative "PRICE" sentiment towards the entity is not expressed until much later. Meanwhile, due to the large number of conjunctions in the sentences, how to track entity state and use target and aspect representations to match the corresponding entity state are the main issues when applying recurrent entity networks to TABSA task. In this work, we propose a novel model architecture for TABSA. Specifically, the proposed model is a recurrent entity network based im-

[1] Hunan University, China, email: zhihaoye.chn@qq.com
[2] Hunan University, China, email: zhiyong.li@hnu.edu.cn
∗ Corresponding author.

provement and application on TABSA task. First, inspired by a prior model [30], we modify it with sentence-level hidden memory, which can represent the sentence-level semantic information of relevant entity at each time step. Then, the target embedding and aspect embedding of the proposed model are fine-tuned through the attention mechanism and all entity states, because we hold the opinion that the representations of target and aspect should take full account of context information rather than use context-independent representation. Finally, the corresponding states matched by target and aspect are input into a softmax classifier.

The main contributions of this work can be summarized as follows.

(1) We propose a novel variant of recurrent entity networks. In our experiments, the effect of our model for TABSA task have greatly improved compared with previous models.

(2) For the first time, recurrent entity networks with sentence-level hidden memory were modified, which can improve the ability of entity state tracking and more effectively solve the issue of long-range dependence when applying recurrent entity networks for TABSA tasks.

(3) For the first time, the proposed model utilize entity states to fine-tune the target embedding and aspect embedding. The representation of target and aspect are generated from highly correlative context rather than using context-independent or randomly initialized embedding.

2 Related Work

Recently, deep learning [22, 14, 29, 26, 28] have achieved great success in (T)ABSA and other natural language processing tasks, and the proposed model is also a deep learning model. Moreover, the proposed model is a variant of a recurrent entity network (EntNet) that was proposed [5] for reasoning-focused machine reading comprehension tasks. However, recurrent entity networks cannot be directly used for TABSA tasks because the original recurrent entity networks are mainly used for reason-focused machine reading comprehension tasks, which input coarse-grained sentences. To adapt recurrent entity networks to the TABSA task, a variant of a recurrent entity networks with delayed memory update [12] was proposed and achieved a good effect on TABSA task. However, their model still does not have effective entity state tracking, loses the semantic information, and ignores the interdependence among the specific target, corresponding aspects, and the context.

Inspired by message passing over graphs [16, 20], this paper proposes adding sentence-level hidden memory to the recurrent entity network to carry out entity state tracking more effectively. In addition, it is like the sentence-state in LSTM proposed in a prior model [30], which also effectively utilizes the sentence-level state; however, unlike their model, which is an alternative LSTM structure for encoding text consisting of a parallel state for each word, we treat the sentence-level state as part of the entity update state in the recurrent entity network.

Furthermore, many works [14, 17] have utilized context independent or randomly initialized vectors to represent the targets and aspects, which loses the semantic information and ignores the interdependence among a specific target, corresponding aspects, and the context. Context-aware embedding has been proposed for targeted aspect-based sentiment analysis [11]. Our fine-tuning module is like this method, but differs in that our model fine-tunes the target embedding and aspect embedding with all the entity states, and it can conduct end-to-end training of the fine-tuning module and sentiment analysis classification module.

3 Methodology

3.1 Framework Overview

The task is framed as a 3-class classification problem. Specifically, given a sentence s, a preidentified set of target entities T and fixed set of aspects A, our model will predict the related sentiment polarity. In TABSA task, a sentence s typically consists of a sequence of words: $w_1, ..., w_i, ..., w_n$ where w_i denotes words interleaved with one or more targets (t) and aspects (a). A specific example can be seen in Figure 1.

We design a neural framework which is capable of tracking and updating the states of entities at the right time with external memory. Specifically, our model maintains a number of memory chains h_j, each entity has the key k_j and the states (h_j) are dynamically updated with word-level information and sentence-level hidden memory q. Finally, the fine-tuned target embedding and aspect embedding are applied to obtain the most consistent state through the memory mechanism, and the final sentiment polarity is classified with a softmax layer.

An overview of the framework is illustrated in Figure 2. A diagram of the recurrent entity networks dynamic memory is shown on the left side of Figure 2, where k is the key of the entity, h is the entity state, and w_i is the i-th word in the text. $f(\theta)$ represents a single memory chain, where θ is the set of trainable parameters. A more detailed illustration of $f(\theta)$ is shown in Figure 3. The bottom right of Figure 2 shows that all the times-step entity states are applied to fine-tune the target embedding and aspect embedding to obtain a context-dependent representation. The upper right side of Figure 2 represents all the entity states and aspects of the last layer, and the target is input into the final classifiers to obtain the corresponding emotion polarity y.

3.2 Dynamic Memory

To better capture and track the entity state, the word-level information and sentence-level hidden memory are used. In addition, each memory chain is updated and controlled is controlled by a gating mechanism, consisting of three component: the "content" term $w_i h^j_{i-1}$ opening for memory chains whose keys match the input; the "location" term $w_i k^j$, which triggers the activation when the content of the entities matches the input and the "delay" term vd^j_i, which causes the gate to open for memory slots whose content matches the input. Specifically, the update gate is calculated as follows:

$$g^j_i = \sigma(w_i h^j_{i-1} + w_i k^j + V d^j_i) \quad (1)$$

where g^j_i is the update gate value for the $j-th$ memory at time i, k_j is the embedding for the j-th entity (key), h^j_{i-1} is the state of the j-th entity (content) at time step $i-1$, and σ is the sigmoid activation function.

$$\tilde{h}^j_i = \phi(U h^j_{i-1} + V w_j + W s_t + S q_t) \quad (2)$$

$$d^j_i = GRU(\tilde{h}^j_i, d^j_{i-1}) \quad (3)$$

where $\tilde{h}^j_i$ is the new candidate value of the memory to be combined with the existing memory h_{j-1}. U, V and W are trainable

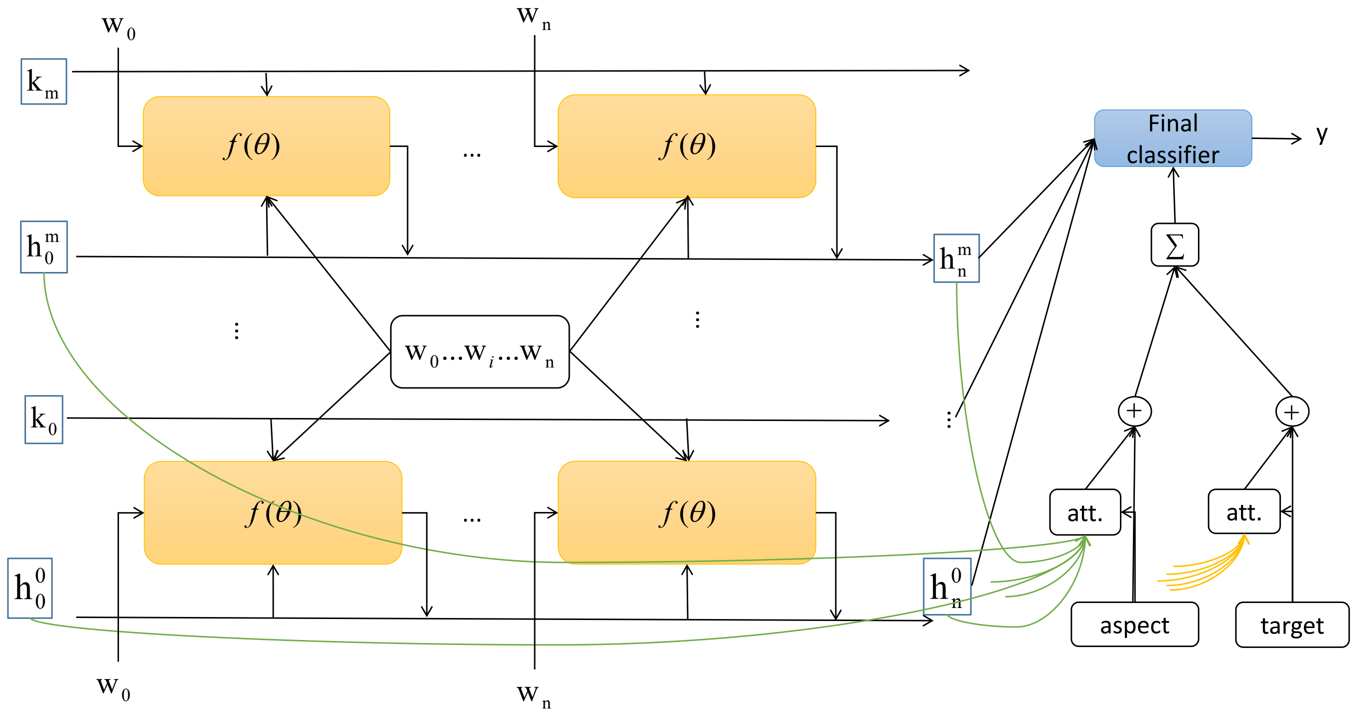

Figure 2. The overview of our framework. (see section 3.1 for more explanations)

weight matrices. q_t is a sentence state that matches the current memory value, for the q_t specific calculations that can be seen in Section 3.3. Like prior work [12], the delay term models decide how and when the gate turns on in the past with a gated recurrent unit(GRU) [4] and how past activations should influence the current one.

$$h_i^j = h_{i-1}^j + g_i^j \odot \tilde{h}_i^j \tag{4}$$

$$h_i^j = \frac{h_i^j}{||h_i^j||} \tag{5}$$

where $||h_i^j||$ denotes the Euclidean norm of h_i^j. The final normalization step allows the model to forget previous information.Because all the information stored in h_i^j is constrained to be of unit length, when new information $\tilde{h}_i^j$ is added to the existing memory h_{i-1}^j, the cosine distance between the original and updated memory decreases, which allows the model to forget information deemed to be out-of-date.

3.3 Sentence-Level Hidden Memory

If only the word-level information is used to update the entity state and sequential information flow in a previous model, it would lead to relatively weaker power in capturing long-range dependencies in TABSA task and the final entity state may not match the specific target or aspect. To make the model more effective for long sentences and employ sentence-level information which can compensate for the loss of entity state information due to sequential information flow over long distances, this paper proposes a novel mechanism that can employ sentence-level hidden memory to help track the entity state. The main idea is to model the hidden memory of sentence that matches the current time step entity state. Specifically, the value of q_t is computed based on all the words $i \in [0..n+1]$.

$$\bar{w} = avg(w_0, w_1...w_i...w_n) \tag{6}$$

where $w_0, w_1...w_i...w_n$ is the word of text. All the word embeddings are averaged to obtain the new sentence embedding representation $\bar{w}$.

$$\hat{f}_t = \sigma(W_1 h_{t-1} + U_1 \bar{w} + b_1) \tag{7}$$

$$\begin{aligned} f_0, f_1...f_i...f_n = \sigma(W_2 h_{t-1} + U_2 w_0 + b_2)... \\ \sigma(W_2 h_{t-1} + U_2 w_i + b_2)...\sigma(W_2 h_{t-1} + U_2 w_n + b_2) \end{aligned} \tag{8}$$

$$\dot{f}_0, \dot{f}_1..., \dot{f}_i..\dot{f}_n = softmax(f_0, f_1...f_i...f_n) \tag{9}$$

Here $f_0, f_1...f_i...f_n$ are the gates controlling information from $w_0, w_1...w_i...w_n$ respectively. After normalization, $\dot{f}_0, \dot{f}_1..., \dot{f}_i..\dot{f}_n$ is obtained with the softmax function. $\hat{f}_t$ is an forget gate at time step t, which can effectively ignore information that is inconsistent with the sentence information. W_1, W_2, U_1, U_2, b_1 and b_2 are the model parameters.

$$q_t = \hat{f}_t \odot h_{t-1} + \sum_{i=1} \dot{f}_i \odot w_i \tag{10}$$

Finally, the sentence-level hidden memory is obtained after adding the weighted sum of each word and the entity state of the previous time step.

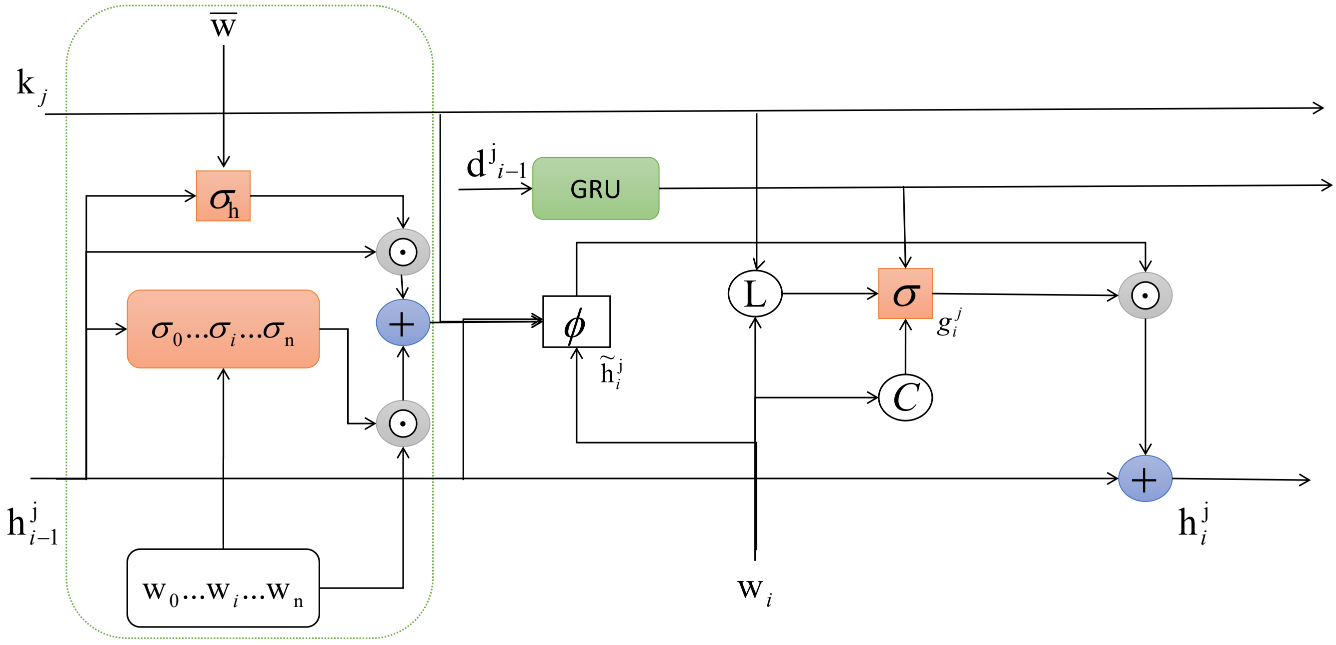

Figure 3. Illustration of the proposed model with a single memory chain at time i, where σ, ϕ and the GRU represent function $\sigma()$, $\phi()$ and GRU model. The dotted box on the right represents the sentence-level hidden memory and the circled nodes L,C, $\odot$ and $+$ depict the location, content terms, Hadamard product, and addition, respectively.

3.4 Fine-Tuning the Target Embedding and Aspect Embedding

Previous works usually utilize context independent or randomly initialized vectors to represent targets and aspects, which loses the semantic information and ignores the interdependence among the specific target, corresponding aspects and the context.

In order to address this issue, in our work, we use an attention mechanism[3] and all the entity states to fine-tune the target embedding and aspect embedding and generate the context-aware representation of the target and aspect.

For aspect embedding, the main idea is that we reconstruct the target embedding of a given sentence according to the highly correlated entity memory h^j_i. h^j_i represents the j-th entity state of at i-th time step, and the attention mechanism can be used to match the context memory information of the target. Then, the original target embedding or aspect embedding can be added to obtain the context-dependent representation. Specifically, for the target embedding t, we have:

$$t_{att} = \sum_{j=0}^{m}\sum_{i=0}^{n} \alpha_{ij} h^j_i \tag{11}$$

where α_j is the similarity of target embedding t and each state of entity j, and m is the total number of entities. α_{ij} is computed as follows:

$$\alpha_{ij} = softmax(th^j_i) \tag{12}$$

$$t' = t + t_{att} \tag{13}$$

where t' is the fine-tuned target embedding. For aspect embedding, the calculation process is similar to fine-tuning the target embedding:

$$a_{att} = \sum_{j=0}^{m}\sum_{i=0}^{n} \alpha_{ij} h^j_i \tag{14}$$

$$\alpha_{ij} = softmax(ah^j_i) \tag{15}$$

$$a' = a + a_{att} \tag{16}$$

where a' is the fine-tuned aspect embedding.

3.5 Final classifier

After aspect embedding and target embedding had been fine-tuned, frisr, the similarity of each key and aspect with target are calculated. Specifically, when detecting aspects, only the target embedding do similarity calculation, and when identifying sentiments for target-aspect pair, both the target embedding and aspect embedding do similarity calculation. In addition, in the case of multi-word aspect expressions (e.g. TRANSIT-LOCATION), we take the mean of the embeddings of the constituent words.

$$p^j = softmax\Big((k^j)^T W_{att} \begin{bmatrix} a' \\ t' \end{bmatrix}\Big) \tag{17}$$

$$u = \sum_j p^j h^j_n \tag{18}$$

where [ ] denotes concatenation, n is sentence length, and W_{att} is a trainable weight matrix. After similarity calculation and weighted sum matching, classify softmax function is used for classification.

Table 1. The result of automatic comparisons of accuracy (Acc.), average F_1 (Avg F_1) and AUC on test data.

Models	Aspect			Sentiment	
	Acc.	Avg F_1	AUC	Avg F_1	AUC
LR	—	39.3	92.4	87.5	90.5
LSTM-Final	—	68.9	89.8	82.0	85.4
LSTM-Loc	—	69.3	89.7	81.9	83.9
LSTM+TA+SA	66.4	76.7	—	86.8	—
EntNet	66.3	69.8	89.5	87.6	89.7
SenticLSTM	67.4	78.2	—	89.3	—
Delayed EntNet	73.5	78.5	94.4	91.0	94.8
Our model†	74.6	79.5	95.4	91.7	95.2
Our model‡	**75.7**	**80.4**	**96.0**	**92.5**	**95.9**

$$y' = softmax(R\phi(Hu + a)) \tag{19}$$

Training is carried out based on cross entropy loss:

$$\mathcal{L} = CrossEntropy(y, y') \tag{20}$$

4 Experiment

4.1 Dataset

To evaluate the capability of the proposed model, the sentihood dataset [19], which is commonly used in the TABSA task, was taken as the training and testing data. The sentihood dataset included 5,215 sentences, where 1,353 of the sentences contained multiple targets and the remaining data contained only one target. In our experiment, this data was used to conduct two experiments, respectively detecting an aspect through giving a sentence and target and identifying the specific sentiment polarity with the sentence and the target-aspect pair.

4.2 Comparison Methods and Evaluation

To evaluate the capability of the proposed model, the evaluation metrics macro-average F_1 and AUC were adopted for aspect detection, and the evaluation metrics accuracy and macro-average AUC were taken for sentiment classification.

We compared the proposed method with the baseline systems presented in prior works [19, 24]. The proposed model and the compared models are described in detail below.

LR: Logistic regression classifier [7] is a classical classification model of machine learning. In the experiment, the n-gram representation of the sentence and the POS tag features were input into the model.

LSTM-based model: LSTM [6] is a commonly used deep learning comparison model. Two different LSTM were used in the comparative experiments: LSTM-Final which is a biLSTM takes the final states as representations; LSTM-Loc, which is a biLSTM that takes the states at the location where the target is mentioned as representations.

LSTM+TA+SA: A biLSTM equipped with complex target and sentence-level attention mechanisms.

SenticLSTM: SenticLSTM is an improved version of LSTM+TA+SA that incorporates the SenticNet external knowledge base [1].

EntNet: EntNet is a recurrent entity network [12], and the proposed model is an improvement on this model, making it more suitable for a TABSA task.

Delayed EntNet: Delayed EntNet is a variant of the recurrent entity network previously proposed [12], and it is the main comparison model.

Our model†: In this experimental model, to test whether sentence-level hidden memory can help entity state tracking and improve the classification ability, sentence-level hidden memory was added based on the original EntNet model and context independent vectors to represent targets and aspects were used.

Our model‡: In this experimental model, based on the original EntNet model, we added sentence-level hidden memory and fine-tune the target embedding and aspect embedding to obtain context dependent embedding and aspect representation.

4.3 Experimental Setup

In the experiment, glove [18] was used as the pre-trained word embedding and the dimension is set to 300. For the model, all the hyper-parameters for the weight matrices were set to be 300 dimensions. For the GRU in Equation (3), the hidden size was set to 300.

For model training and optimization, an adam optimizer [8] was adopted, and the learning rate was set to 0.01. A dropout [21] to the output was applied in the final classifier and input at every step with a rate of 0.2 to avoid overfitting. The model iterated 1,500 epochs in total.

Following prior work [12], the number of memory chains was empirically set to 6, with the keys of two of them set to the same embeddings as the target words, and the other four chains with free key embeddings that were updated during training; therefore, they were free to capture any entities.

4.4 Experimental Results

The experimental results of the proposed model on the sentihood dataset are shown in Table 1. The following conclusions can be made based on an analysis of the experimental results:

The proposed model has a stronger ability of fine-grained sentiment classification than previous models.

As shown in Table 1, the proposed model achieved the best results on both aspect detection and sentiment classification tasks. In particular, compared with the original EntNet and EntNet with delayed memory update, due to the inclusion of sentence level hidden memory and fine-tuning the target embedding and aspect embedding, the proposed model improved the accuracy by nearly 0.02 in the aspect detection and sentiment classification tasks.

Sentence-level hidden memory helps enhance the effect. A comparison of our model† and Delayed EntNet in Table 1 shows that the proposed model had an enhanced effect. This experimental result also proves that the proposed model is effective when adding sentence level hidden memory, which can help entity state tracking.

The proposed model can effectively adapt fine-tuning the target embedding and aspect embedding. As can be seen from the comparison of our model‡ and our model† in Table 1, after fine-tuning the target embedding and aspect embedding, the effect on the dataset sentihood was improved. This proves that the proposed model can better match the sentiment polarity or aspect after target embedding and aspect embedding.

5 Conclusion and Future Work

This work proposes a novel model architecture for TABSA and the experimental results of the model show that it is a great improvement over the previous methods. Specifically, recurrent entity networks are modified with sentence level hidden memory, which can more effectively solve the issue of long-range dependence and improve the ability of entity state tracking. Moreover, entity states are utilized to fine-tune the target embedding and aspect embedding, effectively utilizing the semantic information of the target and aspect. However, we also found that our model consumes more memory and training time. In future work, we will explore how to simplify the model while achieving the same effect. From the perspective of information flow, the proposed model still passes information from one end of the sentence to the other; as a result, the number of time steps scales with the size of the input. In future work, we will explore how to reduce the number of time steps and achieve a better experimental result.

6 Acknowledgements

This work was partially supported by the National Key Research and Development Program of China (No. 2018YFB1308604), National Natural Science Foundation of China (No.61672215, 61976086),and Hunan Science and Technology Innovation Project (No. 2017XK2102).

REFERENCES

[1] Erik Cambria, Soujanya Poria, Rajiv Bajpai, and Björn Schuller, 'Senticnet 4: A semantic resource for sentiment analysis based on conceptual primitives', in *Proceedings of COLING 2016, the 26th international conference on computational linguistics: Technical papers*, pp. 2666–2677, (2016).

[2] Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang, 'Recurrent attention network on memory for aspect sentiment analysis', in *Proceedings of the 2017 conference on empirical methods in natural language processing*, pp. 452–461, (2017).

[3] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, 'Learning phrase representations using rnn encoder-decoder for statistical machine translation', *arXiv preprint arXiv:1406.1078*, (2014).

[4] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio, 'Empirical evaluation of gated recurrent neural networks on sequence modeling', *arXiv preprint arXiv:1412.3555*, (2014).

[5] Mikael Henaff, Jason Weston, Arthur Szlam, Antoine Bordes, and Yann LeCun, 'Tracking the world state with recurrent entity networks', *arXiv preprint arXiv:1612.03969*, (2016).

[6] Sepp Hochreiter and Jürgen Schmidhuber, 'Long short-term memory', *Neural computation*, **9**(8), 1735–1780, (1997).

[7] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant, *Applied logistic regression*, volume 398, John Wiley & Sons, 2013.

[8] Diederik P Kingma and Jimmy Ba, 'Adam: A method for stochastic optimization', *arXiv preprint arXiv:1412.6980*, (2014).

[9] Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad, '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).

[10] Xin Li and Wai Lam, 'Deep multi-task learning for aspect term extraction with memory interaction', in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2886–2892, (2017).

[11] Bin Liang, Jiachen Du, Ruifeng Xu, Binyang Li, and Hejiao Huang, 'Context-aware embedding for targeted aspect-based sentiment analysis', *arXiv preprint arXiv:1906.06945*, (2019).

[12] Fei Liu, Trevor Cohn, and Timothy Baldwin, 'Recurrent entity networks with delayed memory update for targeted aspect-based sentiment analysis', *arXiv preprint arXiv:1804.11019*, (2018).

[13] Qiao Liu, Haibin Zhang, Yifu Zeng, Ziqi Huang, and Zufeng Wu, 'Content attention model for aspect based sentiment analysis', in *Proceedings of the 2018 World Wide Web Conference*, pp. 1023–1032. International World Wide Web Conferences Steering Committee, (2018).

[14] Yukun Ma, Haiyun Peng, and Erik Cambria, 'Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm', in *Thirty-Second AAAI Conference on Artificial Intelligence*, (2018).

[15] Navonil Majumder, Soujanya Poria, Alexander Gelbukh, Md Shad Akhtar, Erik Cambria, and Asif Ekbal, 'Iarm: Inter-aspect relation modeling with memory networks in aspect-based sentiment analysis', in *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 3402–3411, (2018).

[16] Kevin P Murphy, Yair Weiss, and Michael I Jordan, 'Loopy belief propagation for approximate inference: An empirical study', in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pp. 467–475. Morgan Kaufmann Publishers Inc., (1999).

[17] Minlong Peng, Qi Zhang, Yu-gang Jiang, and Xuan-Jing Huang, 'Cross-domain sentiment classification with target domain specific information', in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2505–2513, (2018).

[18] Jeffrey Pennington, Richard Socher, and Christopher Manning, 'Glove: Global vectors for word representation', in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, (2014).

[19] Marzieh Saeidi, Guillaume Bouchard, Maria Liakata, and Sebastian Riedel, 'Sentihood: Targeted aspect based sentiment analysis dataset for urban neighbourhoods', *arXiv preprint arXiv:1610.03771*, (2016).

[20] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini, 'The graph neural network model', *IEEE Transactions on Neural Networks*, **20**(1), 61–80, (2008).

[21] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, 'Dropout: a simple way to prevent neural networks from overfitting', *The journal of machine learning research*, **15**(1), 1929–1958, (2014).

[22] Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu, 'Effective lstms for target-dependent sentiment classification', *arXiv preprint arXiv:1512.01100*, (2015).

[23] Duyu Tang, Bing Qin, and Ting Liu, 'Aspect level sentiment classification with deep memory network', *arXiv preprint arXiv:1605.08900*, (2016).

[24] Joachim Wagner, Piyush Arora, Santiago Cortes, Utsab Barman, Dasha Bogdanova, Jennifer Foster, and Lamia Tounsi, 'Dcu: Aspect-based polarity classification for semeval task 4', (2014).

[25] Shuai Wang, Sahisnu Mazumder, Bing Liu, Mianwei Zhou, and Yi Chang, 'Target-sensitive memory networks for aspect sentiment classification', in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 957–967, (2018).

[26] Yequan Wang, Minlie Huang, Li Zhao, 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).

[27] Jason Weston, Sumit Chopra, and Antoine Bordes, 'Memory networks', *arXiv preprint arXiv:1410.3916*, (2014).

[28] Wei Xue and Tao Li, 'Aspect based sentiment analysis with gated convolutional networks', *arXiv preprint arXiv:1805.07043*, (2018).
[29] Min Yang, Wenting Tu, Jingxuan Wang, Fei Xu, and Xiaojun Chen, 'Attention based lstm for target dependent sentiment classification', in *Thirty-First AAAI Conference on Artificial Intelligence*, (2017).
[30] Yue Zhang, Qi Liu, and Linfeng Song, 'Sentence-state lstm for text representation', *arXiv preprint arXiv:1805.02474*, (2018).
[31] Peisong Zhu and Tieyun Qian, 'Enhanced aspect level sentiment classification with auxiliary memory', in *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 1077–1087, (2018).