Aspect-based Sentiment Classification with Aspect-specific Graph Convolutional Networks

Chen Zhang[1], Qiuchi Li[2], Dawei Song[1]

[1]Beijing Institute of Technology, China. {czhang, dwsong}@bit.edu.cn

[2]University of Padua, Italy. {qiuchili}@dei.unipd.it

EMNLP 2019, Hong Kong, China

UNIVERSITÀ DEGLI STUDI DI PADOVA

Content

- Problem Formulation
- Related Work
- Motivation
- Proposed Model – ASGCN
- Experiments
- Discussion
- Conclusions & Future Work

Content

- Problem Formulation
- Related Work
- Motivation
- Proposed Model – ASGCN
- Experiments
- Discussion
- Conclusions & Future Work

Problem Formulation

- **Aspect-based Sentiment Classification** (ABSC): aiming at identifying the sentiment polarities of aspect terms explicitly given in sentences.

From the speed to the multi-touch gestures this operating system beats Windows easily.

- Aspect terms (a.k.a. opinion targets): operating system, Windows
- Corresponding sentiment polarities: positive, negative

Content

- Problem Formulation
- Related Work
- Motivation
- Proposed Model – ASGCN
- Experiments
- Discussion
- Conclusions & Future Work

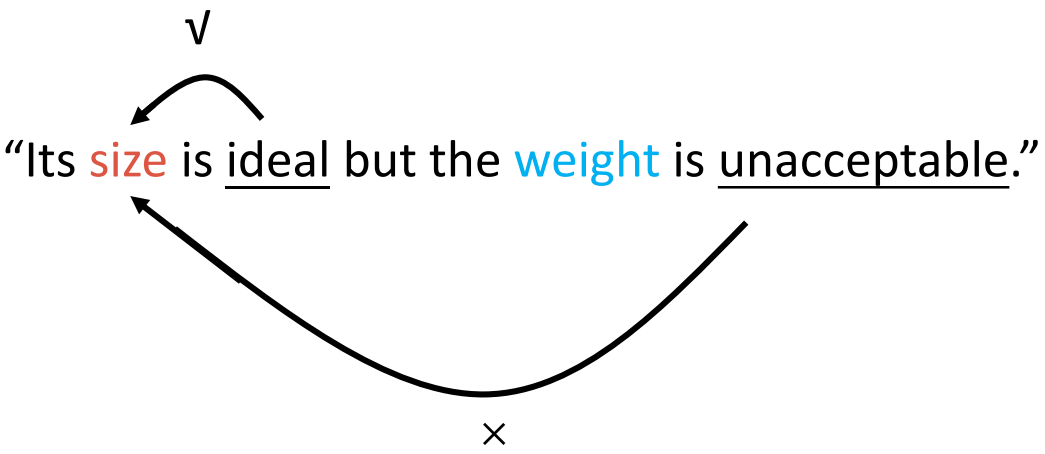
Related Work

- Feature-based approaches
  - Manual features + SVM (Jiang et al., 2011)
- Neural network-based approaches
  - Embedding oriented features (Vo and Zhang, 2015)
  - Recursive neural networks (Dong et al., 2014)
- End-to-end approaches
  - Mainly based on recurrent neural network and attention mechanism
  - TD-LSTM (Tang et al., 2016a)
  - MemNet (Tang et al., 2016b)
  - TNet (Li et al., 2018)
  - etc.

Limitations of the SoTA

- **Observation 1:**

The attention mechanism commonly used in previous ABSC models may result in a given aspect mistakenly attending to *syntactically unrelated context words as descriptors*



Limitations of the SoTA

- **Observation 2:**

These models are inadequate to determine sentiments depicted by multiple words with *long-range dependencies*

"The staff *should be a bit more friendly*."

Long-range dependency

Limitations of the SoTA

Sole attention mechanism (at semantic level) is not enough !

Content

- Problem Formulation
- Related Work
- Motivation
- Proposed Model – ASGCN
- Experiments
- Discussion
- Conclusions & Future Work

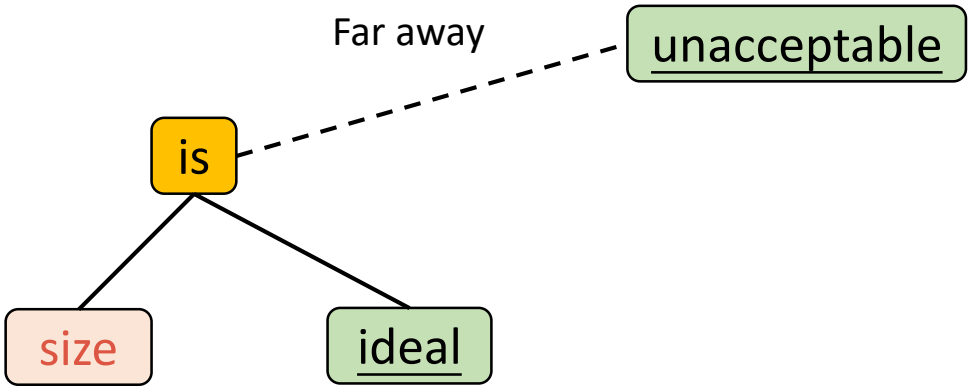
Motivation

Sole attention mechanism (at semantic level) is not enough !

How about incorporating with dependency trees (at syntax level)?

Motivation

"Its size is ideal but the weight is unacceptable."

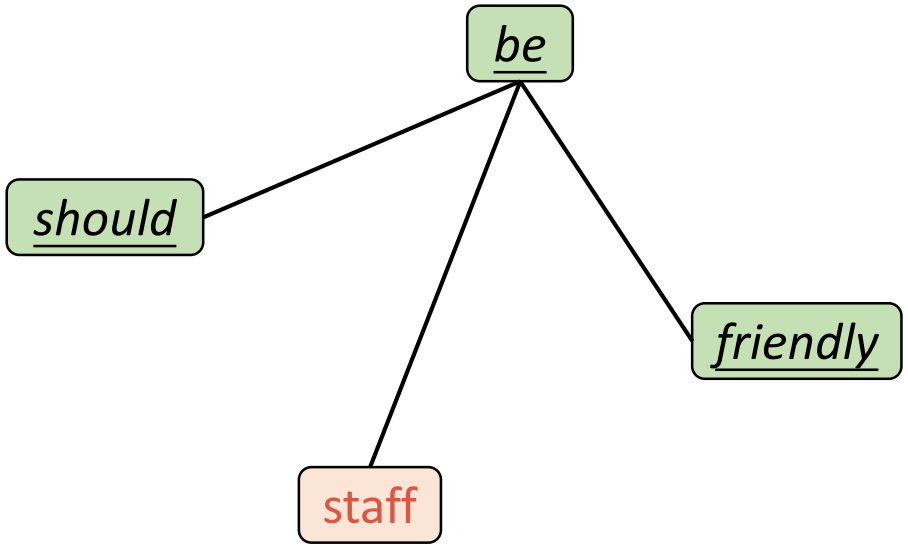


Dependency Tree

Syntactically unrelated words can be ignored by the aspect terms via distance computing

Motivation

"The staff *should be a bit more friendly*."



Dependency Tree

Long-range dependencies can be shortened

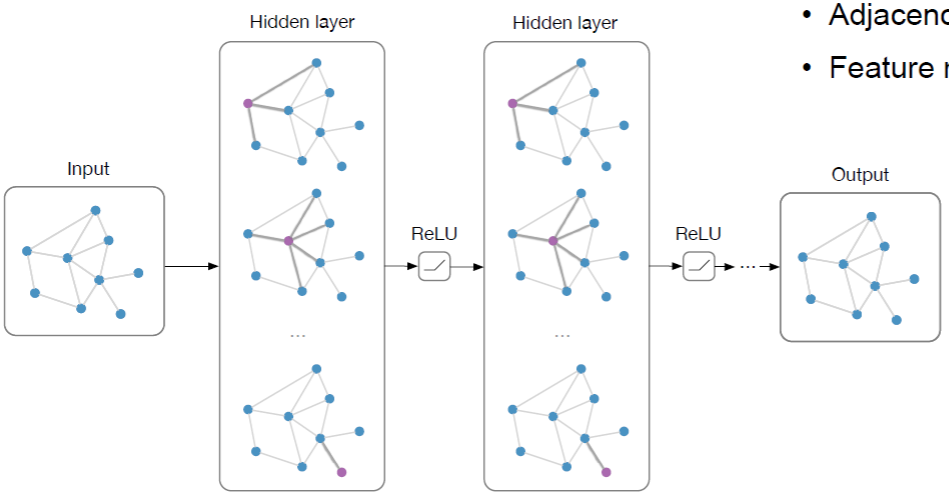
Motivation

- Dependency trees can
  - draw long sequences of words that are syntactically relevant closer
  - keep irrelevant component words far away from aspect terms
- but they are insufficient to
  - capture words' semantics
- We propose to build Graph Convolutional Networks (GCNs) over dependency trees to
  - draw syntactically relevant words *a step closer* to the aspect
  - exploit both (latent) semantics and syntactic information

GNNs

Graph Neural Networks (GNNs)

The bigger picture:



Notation: $\mathcal{G} = (\mathbf{A}, \mathbf{X})$

- Adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$
- Feature matrix $\mathbf{X} \in \mathbb{R}^{N \times F}$

Main idea: Pass messages between pairs of nodes & agglomerate

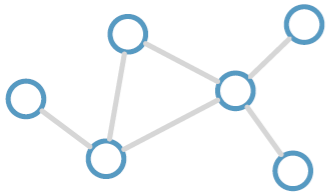
Slides from "Structured deep models: Deep learning on graphs and beyond", Thomas Kipf, 2018.

GCNs

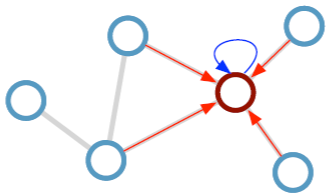
Graph convolutional networks (GCNs)

Kipf & Welling (ICLR 2017), related previous works by Duvenaud et al. (NIPS 2015) and Li et al. (ICLR 2016)

Consider this undirected graph:



Calculate update for node in red:



Update rule: $$\mathbf{h}_i^{(l+1)} = \sigma\left(\mathbf{h}_i^{(l)}\mathbf{W}_0^{(l)} + \sum_{j\in\mathcal{N}_i}\frac{1}{c_{ij}}\mathbf{h}_j^{(l)}\mathbf{W}_1^{(l)}\right)$$

Scalability: subsample messages [Hamilton et al., NIPS 2017]

$\mathcal{N}_i$: neighbor indices

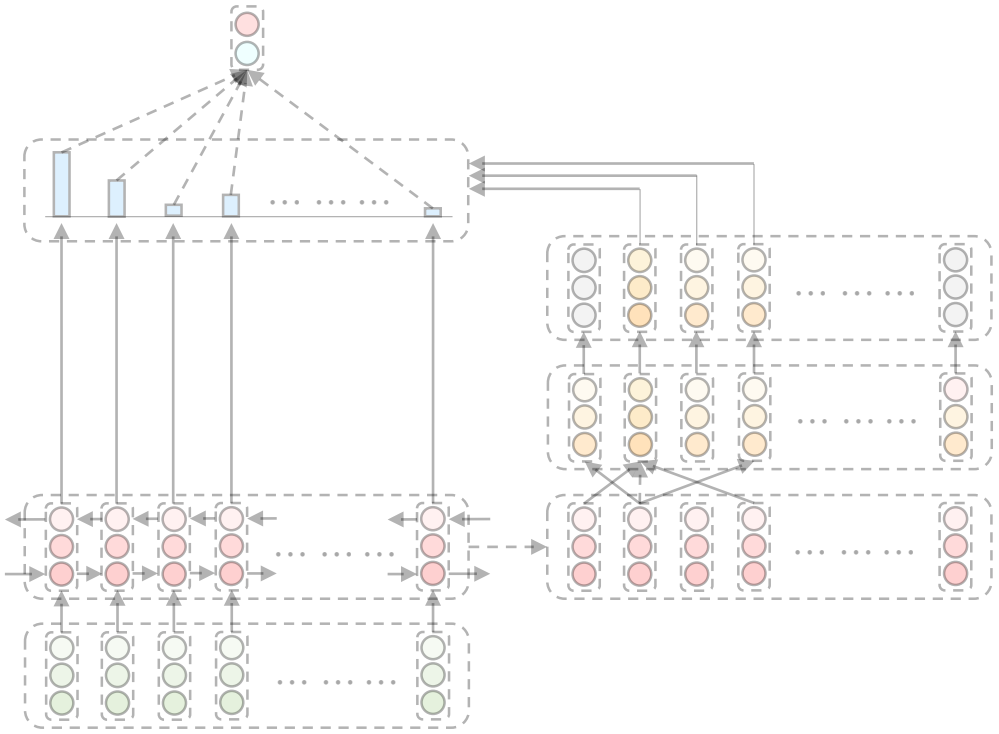
c_{ij} : norm. constant (fixed/trainable)

Slides from “Structured deep models: Deep learning on graphs and beyond”, Thomas Kipf, 2018.

Content

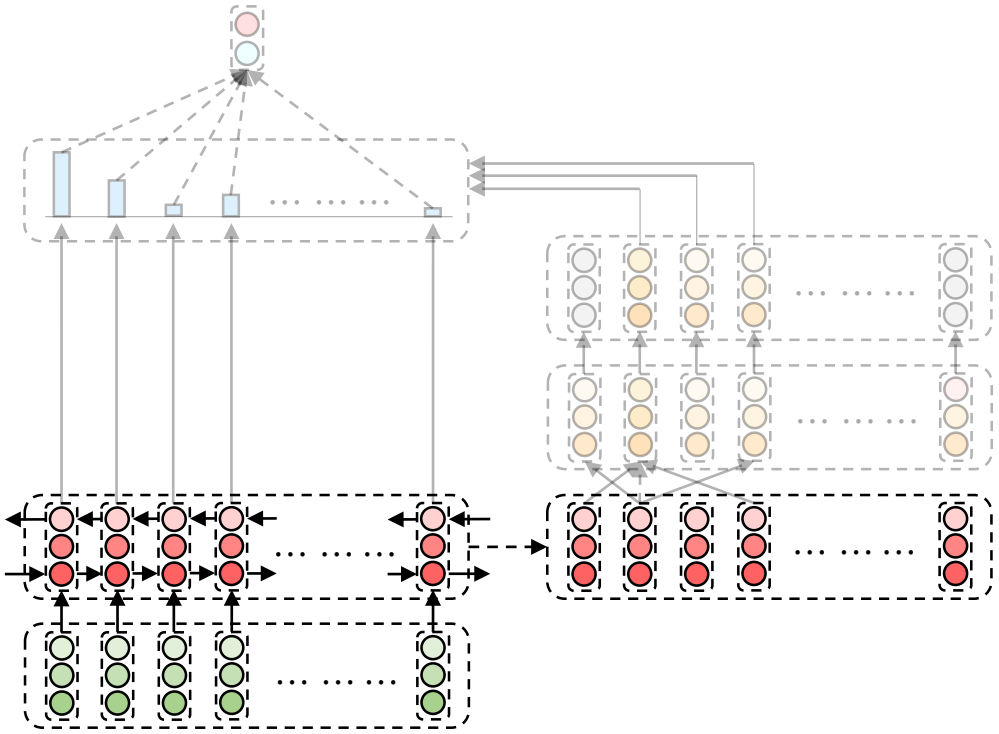
- Problem Formulation
- Related Work
- Motivation
- Proposed Model – ASGCN
- Experiments
- Discussion
- Conclusions & Future Work

Aspect-Specific GCN (ASGCN) - Overview



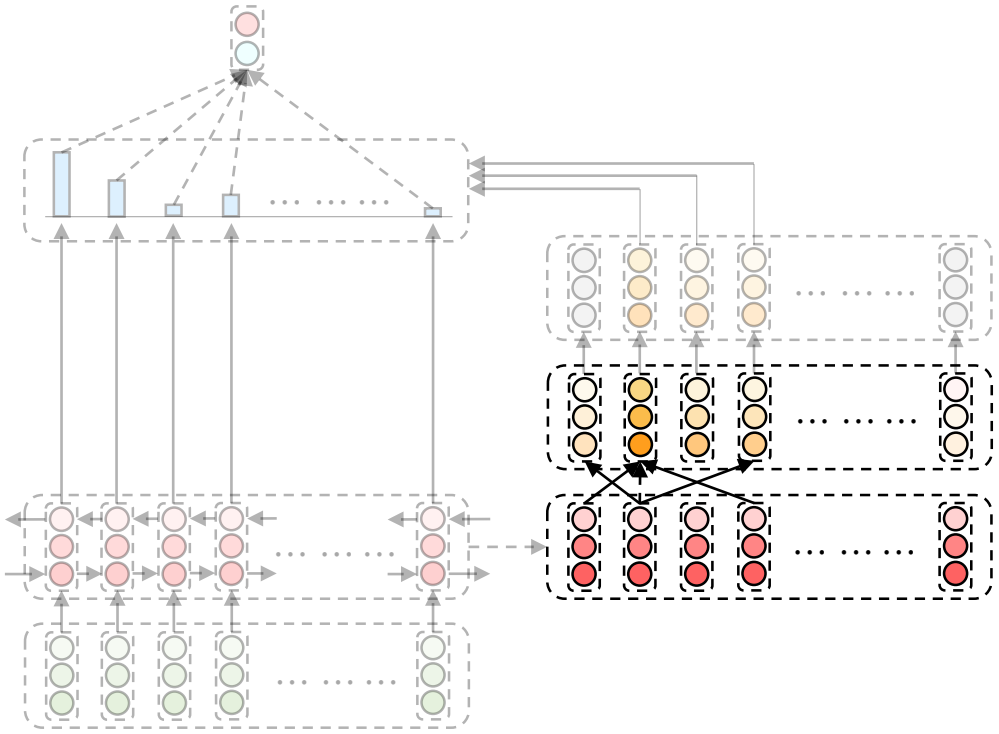
- ASGCN is composed of
  - Word embeddings
  - Bidirectional LSTM
  - GCNs
  - Aspect-specific masking
  - Attention
- Dependency tree is regarded
  - As a graph (ASGCN-DG) or
  - As a tree (ASGCN-DT)

Embeddings & BiLSTM



- With the word embeddings of a sentence, a bidirectional LSTM is constructed to produce hidden state vectors $\mathbf{H}^c$
- Following Zhang et al., 2018, we make nodes on the graph aware of context by initializing the nodes representation to $\mathbf{H}^c$, i.e. $\mathbf{H}^0 = \mathbf{H}^c$

GCNs



- Graph convolution at l–th layer (totally L layers)

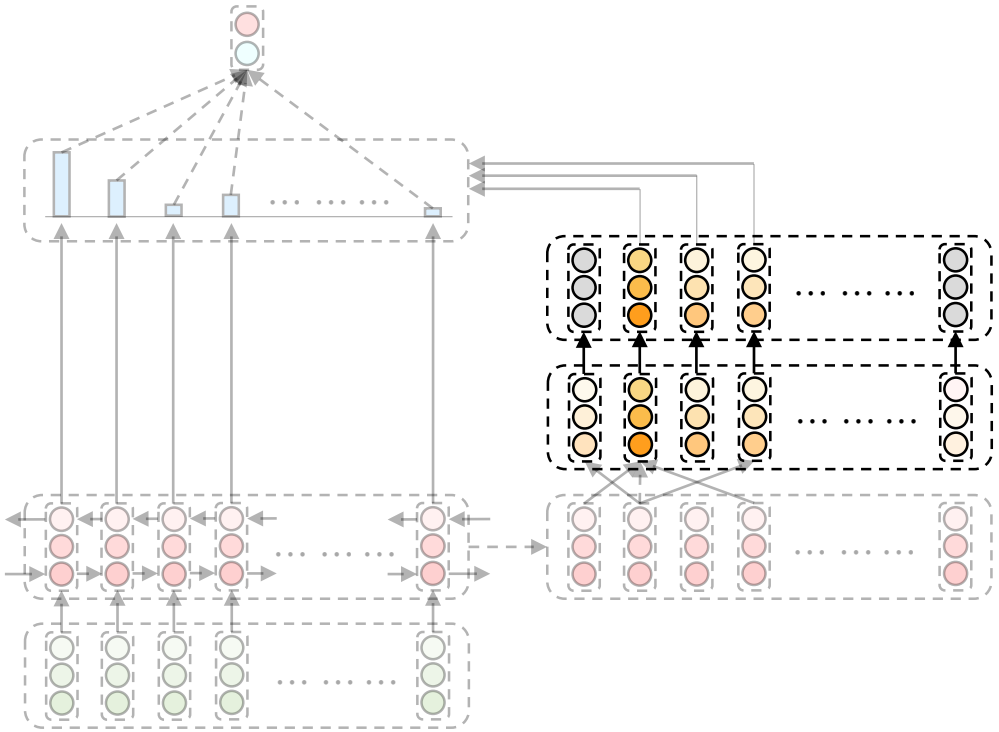
$$\tilde{\mathbf{h}}_i^l = \sum_{j=1}^{n} \mathbf{A}_{ij} \mathbf{W}^l \mathbf{g}_j^{l-1}$$

$$\mathbf{h}_i^l = \mathrm{ReLU}(\tilde{\mathbf{h}}_i^l / (d_i + 1) + \mathbf{b}^l)$$

- We also incorporate position weights, which is commonly used in ABSC models, with GCNs

$$\mathbf{g}_i^l = \mathcal{F}(\mathbf{h}_i^l)$$

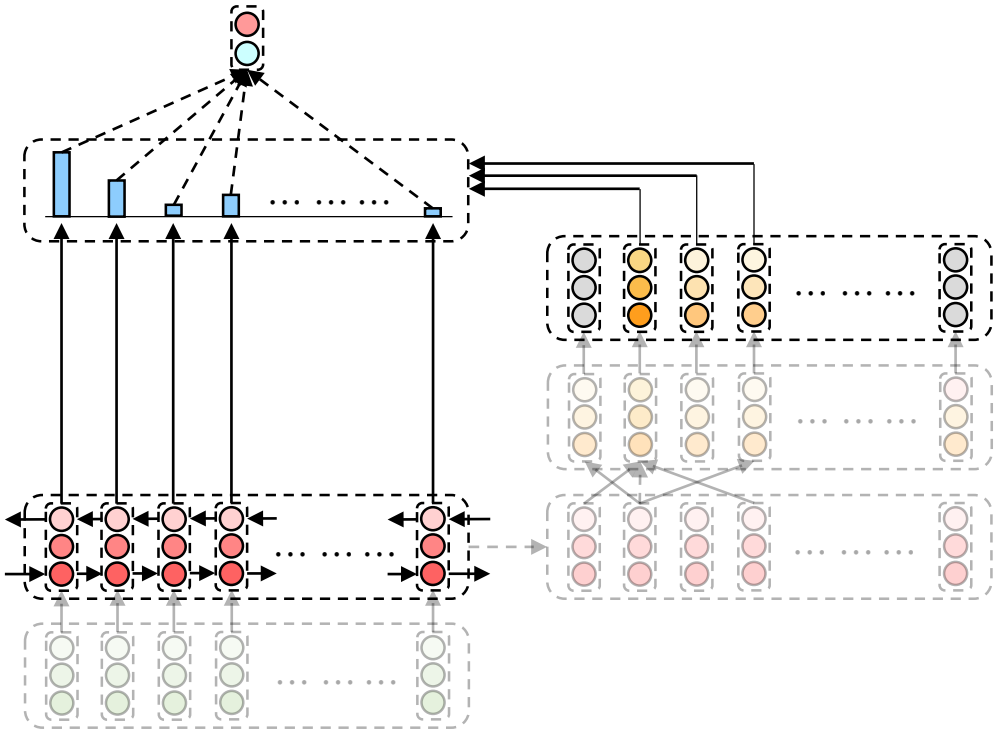
Aspect-specific Masking



- Aspect-specific masking is proposed to get aspect-oriented features

$$\mathbf{h}_t^L = \mathbf{0} \quad 1 \leq t < \tau + 1, \tau + m < t \leq n$$

Attention



- Attention scores are computed based on inner product (to facilitate the masking mechanism)

$$\beta_t = \sum_{i=1}^{n} \mathbf{h}_t^{c\top} \mathbf{h}_i^L = \sum_{i=\tau+1}^{\tau+m} \mathbf{h}_t^{c\top} \mathbf{h}_i^L$$

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

- Final representation is computed as

$$\mathbf{r} = \sum_{t=1}^{n} \alpha_t \mathbf{h}_t^c$$

$$\mathbf{p} = \text{softmax}(\mathbf{W}_p \mathbf{r} + \mathbf{b}_p)$$

Content

- Problem Formulation
- Related Work
- Motivation
- Proposed Model – ASGCN
- Experiments
- Discussion
- Conclusions & Future Work

Datasets & Settings

- **TWITTER**: built by Dong et al. (2014), containing twitter posts
- **LAP14**, **REST14**, **REST15**, **REST16**: respectively from SemEval 2014 task 4, SemEval 2015 task 12 and SemEval 2016 task 5 (Pontiki et al., 2014, 2015, 2016), consisting of data from two categories, i.e. laptop and restaurant
- The number of GCN layers is set to 2, which is the best-performing depth in pilot studies (more illustration later)
- More parameter setting details could be found in paper
- ***Accuracy*** and ***Macro-Averaged F1***

Comparison

- Baselines
  - SVM (Kiritchenko et al., 2014)
  - LSTM (Tang et al., 2016a)
  - MemNet (Tang et al., 2016b)
  - AOA (Huang et al., 2018)
  - IAN (Ma et al., 2017)
  - TNet-LF (Li et al., 2018) (***state-of-the-art***)
- Variants
  - ASCNN, which replaces 2-layer GCN with 2-layer CNN in ASGCN

Comparison

Model	TWITTER		LAP14		REST14		REST15		REST16	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
SVM	63.40♯	63.30♯	70.49♮	N/A	80.16♮	N/A	N/A	N/A	N/A	N/A
LSTM	69.56	67.70	69.28	63.09	78.13	67.47	77.37	55.17	86.80	63.88
MemNet	71.48	69.90	70.64	65.17	79.61	69.64	77.31	58.28	85.44	65.99
AOA	72.30	70.20	72.62	67.52	79.97	70.42	78.17	57.02	87.50	66.21
IAN	**72.50**	**70.81**	72.05	67.38	79.26	70.09	78.54	52.65	84.74	55.21
TNet-LF	**72.98**	**71.43**	**74.61**	**70.14**	80.42	71.03	78.47	59.47	**89.07**	**70.43**
ASCNN	71.05	69.45	72.62	66.72	**81.73**	**73.10**	78.47	58.90	87.39	64.56
ASGCN-DT	71.53	69.68	74.14†	69.24†	**80.86**‡	**72.19**‡	**79.34**†‡	**60.78**†‡	88.69†	66.64†
ASGCN-DG	72.15†	70.40†	**75.55**†‡	**71.05**†‡	80.77‡	72.02‡	**79.89**†‡	**61.89**†‡	**88.99**†	**67.48**†

Table 2: Model comparison results (%). Average accuracy and macro-F1 score over 3 runs with random initialization. The best two results with each dataset are in bold. The results with ♮ are retrieved from the original papers and the results with ♯ are retrieved from Dong et al. (2014). The marker † refers $p < 0.05$ by comparing with ASCNN in paired t-test and the marker ‡ refers $p < 0.05$ by comparing with TNet-LF in paired t-test.

Comparison

Model	TWITTER		LAP14		REST14		REST15		REST16	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
SVM	63.40♯	63.30♯	70.49♮	N/A	80.16♮	N/A	N/A	N/A	N/A	N/A
LSTM	69.56	67.70	69.28	63.09	78.13	67.47	77.37	55.17	86.80	63.88
MemNet	71.48	69.90	70.64	65.17	79.61	69.64	77.31	58.28	85.44	65.99
AOA	72.30	70.20	72.62	67.52	79.97	70.42	78.17	57.02	87.50	66.21
IAN	**72.50**	**70.81**	72.05	67.38	79.26	70.09	78.54	52.65	84.74	55.21
TNet-LF	**72.98**	**71.43**	**74.61**	**70.14**	80.42	71.03	78.47	59.47	**89.07**	**70.43**
ASCNN	71.05	69.45	72.62	66.72	**81.73**	**73.10**	78.47	58.90	87.39	64.56
ASGCN-DT	71.53	69.68	74.14†	69.24†	**80.86**‡	**72.19**‡	**79.34**†‡	**60.78**†‡	88.69†	66.64†
ASGCN-DG	72.15†	70.40†	**75.55**†‡	**71.05**†‡	80.77‡	72.02‡	**79.89**†‡	**61.89**†‡	**88.99**†	**67.48**†

Table 2: Model comparison results (%). Average accuracy and macro-F1 score over 3 runs with random initialization. The best two results with each dataset are in bold. The results with ♮ are retrieved from the original papers and the results with ♯ are retrieved from Dong et al. (2014). The marker † refers $p < 0.05$ by comparing with ASCNN in paired t-test and the marker ‡ refers $p < 0.05$ by comparing with TNet-LF in paired t-test.

Comparison

Model	TWITTER		LAP14		REST14		REST15		REST16	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
SVM	63.40♯	63.30♯	70.49♮	N/A	80.16♮	N/A	N/A	N/A	N/A	N/A
LSTM	69.56	67.70	69.28	63.09	78.13	67.47	77.37	55.17	86.80	63.88
MemNet	71.48	69.90	70.64	65.17	79.61	69.64	77.31	58.28	85.44	65.99
AOA	72.30	70.20	72.62	67.52	79.97	70.42	78.17	57.02	87.50	66.21
IAN	**72.50**	**70.81**	72.05	67.38	79.26	70.09	78.54	52.65	84.74	55.21
TNet-LF	**72.98**	**71.43**	**74.61**	**70.14**	80.42	71.03	78.47	59.47	**89.07**	**70.43**
ASCNN	71.05	69.45	72.62	66.72	**81.73**	**73.10**	78.47	58.90	87.39	64.56
ASGCN-DT	71.53	69.68	74.14†	69.24†	**80.86**‡	**72.19**‡	**79.34**†‡	**60.78**†‡	88.69†	66.64†
ASGCN-DG	72.15†	70.40†	**75.55**†‡	**71.05**†‡	80.77‡	72.02‡	**79.89**†‡	**61.89**†‡	**88.99**†	**67.48**†

Table 2: Model comparison results (%). Average accuracy and macro-F1 score over 3 runs with random initialization. The best two results with each dataset are in bold. The results with ♮ are retrieved from the original papers and the results with ♯ are retrieved from Dong et al. (2014). The marker † refers $p < 0.05$ by comparing with ASCNN in paired t-test and the marker ‡ refers $p < 0.05$ by comparing with TNet-LF in paired t-test.

Comparison

- Results
  - ASGCN shows a competitive performance against strong baselines
  - ASGCN-DG is generally better than ASGCN-DT
  - ASGCN is better at *capturing long-range word dependencies* than ASCNN
  - ASGCN *performs less well on less grammatical datasets* such as **TWITTER**
- Implications
  - If we consider taking dependency trees as directed graph (e.g., ASGCN-DT), we'd better also consider the *edge label information*
  - We need more robust dependency parsers to *reduce the effect of error propagation*

Ablation Study

- w/o pos. : without position weight
- w/o mask: without aspect-specific masking
- w/o GCN: skip GCN layers

Model	TWITTER		LAP14		REST14		REST15		REST16	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
BiLSTM+Attn	71.24	69.55	72.83	67.82	79.85	70.03	78.97	58.18	87.28	68.18
ASGCN-DG	72.15	70.40	75.55	71.05	80.77	72.02	79.89	61.89	88.99	67.48
ASGCN-DG w/o pos.	72.69	70.59	73.93	69.63	81.22	72.94	79.58	61.55	88.04	66.63
ASGCN-DG w/o mask	72.64	70.63	72.05	66.56	79.02	68.29	77.80	57.51	86.36	61.41
ASGCN-DG w/o GCN	71.92	70.63	73.51	68.83	79.40	69.43	79.40	61.18	87.55	66.19

Table 3: Ablation study results (%). Accuracy and macro-F1 scores are the average value over 3 runs with random initialization.

Ablation Study

- w/o pos. : without position weight
- w/o mask: without aspect-specific masking
- w/o GCN: skip GCN layers

Model	TWITTER		LAP14		REST14		REST15		REST16	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
BiLSTM+Attn	71.24	69.55	72.83	67.82	79.85	70.03	78.97	58.18	87.28	68.18
ASGCN-DG	72.15	70.40	75.55	71.05	80.77	72.02	79.89	61.89	88.99	67.48
ASGCN-DG w/o pos.	72.69	70.59	73.93	69.63	81.22	72.94	79.58	61.55	88.04	66.63
ASGCN-DG w/o mask	72.64	70.63	72.05	66.56	79.02	68.29	77.80	57.51	86.36	61.41
ASGCN-DG w/o GCN	71.92	70.63	73.51	68.83	79.40	69.43	79.40	61.18	87.55	66.19

Table 3: Ablation study results (%). Accuracy and macro-F1 scores are the average value over 3 runs with random initialization.

Ablation Study

- Results
  - Position weight is not working for all data
  - Aspect-specific masking has positive effects
  - Removal of GCN brings significant drops in performance (except for **TWITTER**)
- Implications
  - The integration of position weights is not crucial for less grammatical sentences
  - Aspect-specific masking mechanism is important in ASGCN

Case Study

Model	Aspect	Attention visualization	Prediction	Label
MemNet	food	great food but the service was dreadful !	negative✗	positive
	staff	The staff should be a bit more friendly .	positive✗	negative
	Windows 8	Did not enjoy the new Windows 8 and touchscreen functions .	positive✗	negative
IAN	food	great food but the service was dreadful !	positive✓	positive
	staff	The staff should be a bit more friendly .	positive✗	negative
	Windows 8	Did not enjoy the new Windows 8 and touchscreen functions .	neutral✗	negative
ASCNN	food	great food but the service was dreadful !	positive✓	positive
	staff	The staff should be a bit more friendly .	negative✓	negative
	Windows 8	Did not enjoy the new Windows 8 and touchscreen functions .	positive✗	negative
ASGCN-DG	food	great food but the service was dreadful !	positive✓	positive
	staff	The staff should be a bit more friendly .	negative✓	negative
	Windows 8	Did not enjoy the new Windows 8 and touchscreen functions .	negative✓	negative

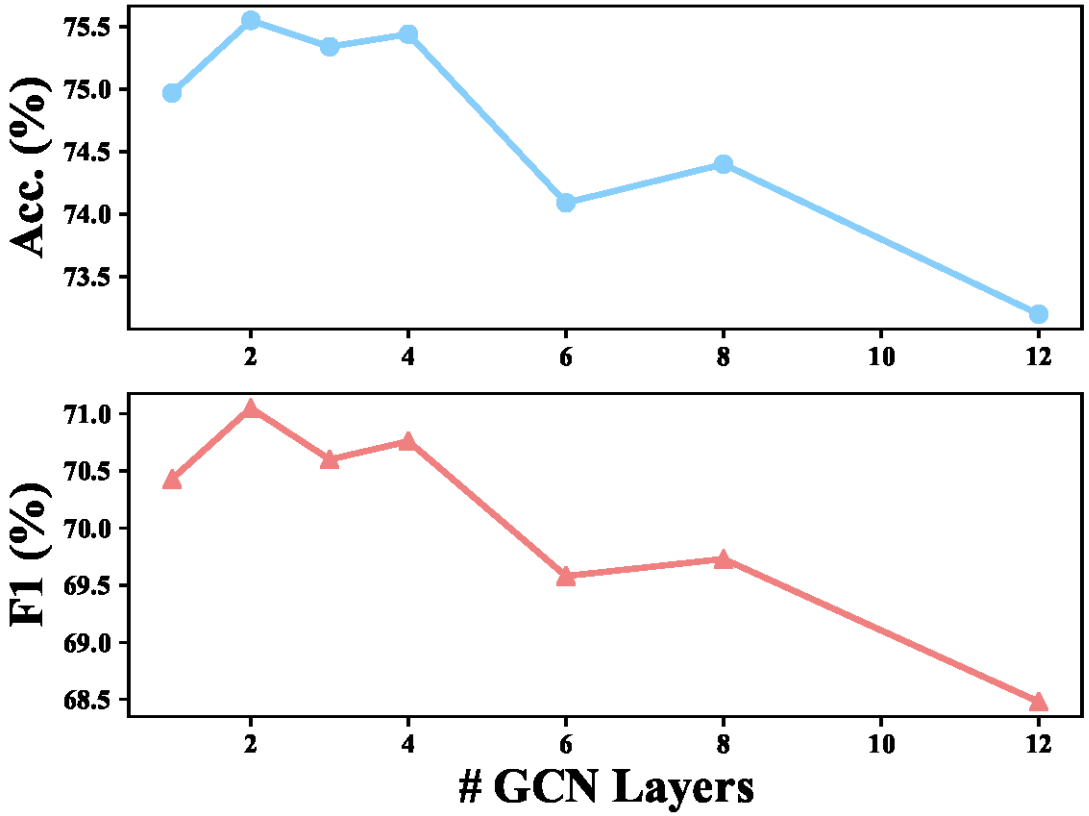
Table 4: Case study. Visualization of attention scores from MemNet, IAN, ASCNN and ASGCN-DG on testing examples, along with their predictions and correspondingly, golden labels. The marker ✓ indicates correct prediction while the marker ✗ indicates incorrect prediction.

Content

- Problem Formulation
- Related Work
- Motivation
- Proposed Model – ASGCN
- Experiments
- Discussion
- Conclusions & Future Work

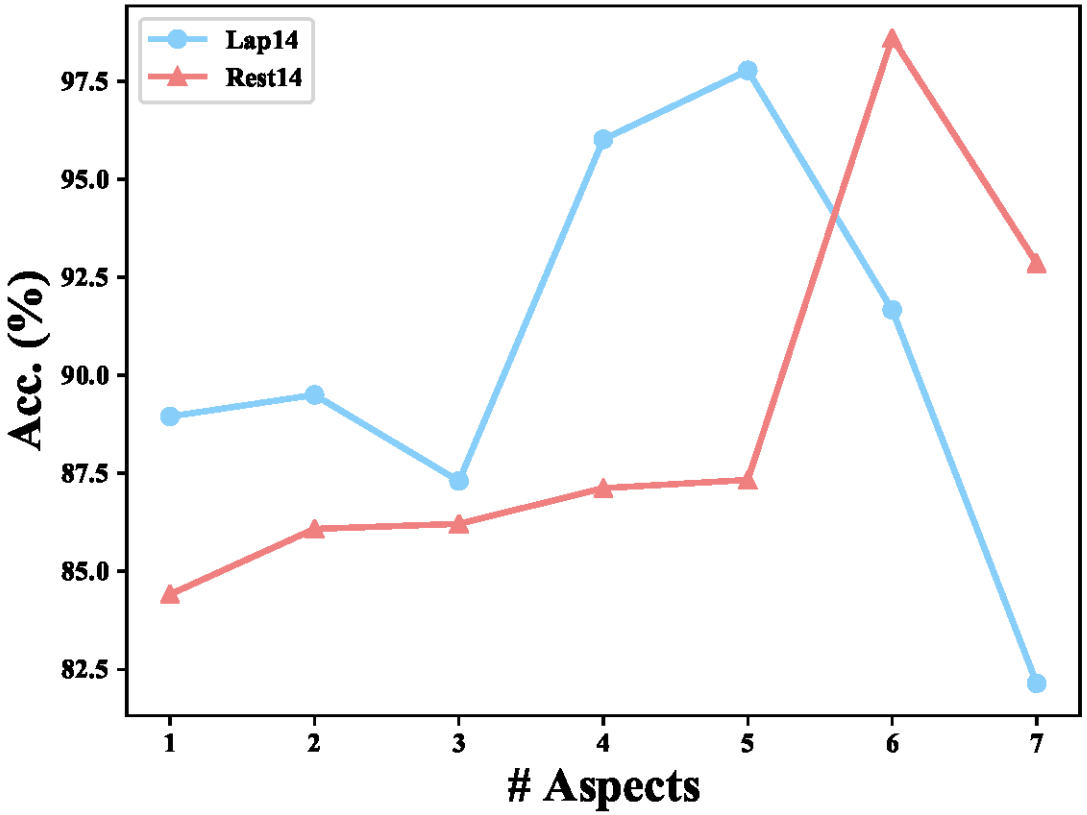
Impact of GCN Layers

- We could see that ***2*** is the best choice under our settings



Effect of Multiple Aspects

- ASGCN shows a ***high variance*** with respect to sentences with different number of aspect terms



Content

- Problem Formulation
- Related Work
- Motivation
- Proposed Model – ASGCN
- Experiments
- Discussion
- Conclusions & Future Work

Conclusions & Future Work

- Conclusions
  - GCNs over dependency trees bring benefit to the overall performance.
  - ASGCN is less effective for **ungrammatical contents** owing to **error propagation** of dependency trees
  - GCNs **with graphs** are better than those **with trees**
  - ASGCN is less robust to **multi-aspect scenarios**
- Future work
  - Reduce errors of dependency parsers via ***joint modelling***
  - Incorporate ***edge information*** of dependency trees
  - Reduce prediction variance via ***judging multiple aspects' polarities*** at the same time

The end

Thanks!

Q&A

https://github.com/GeneZC/ASGCN

https://arxiv.org/abs/1909.03477