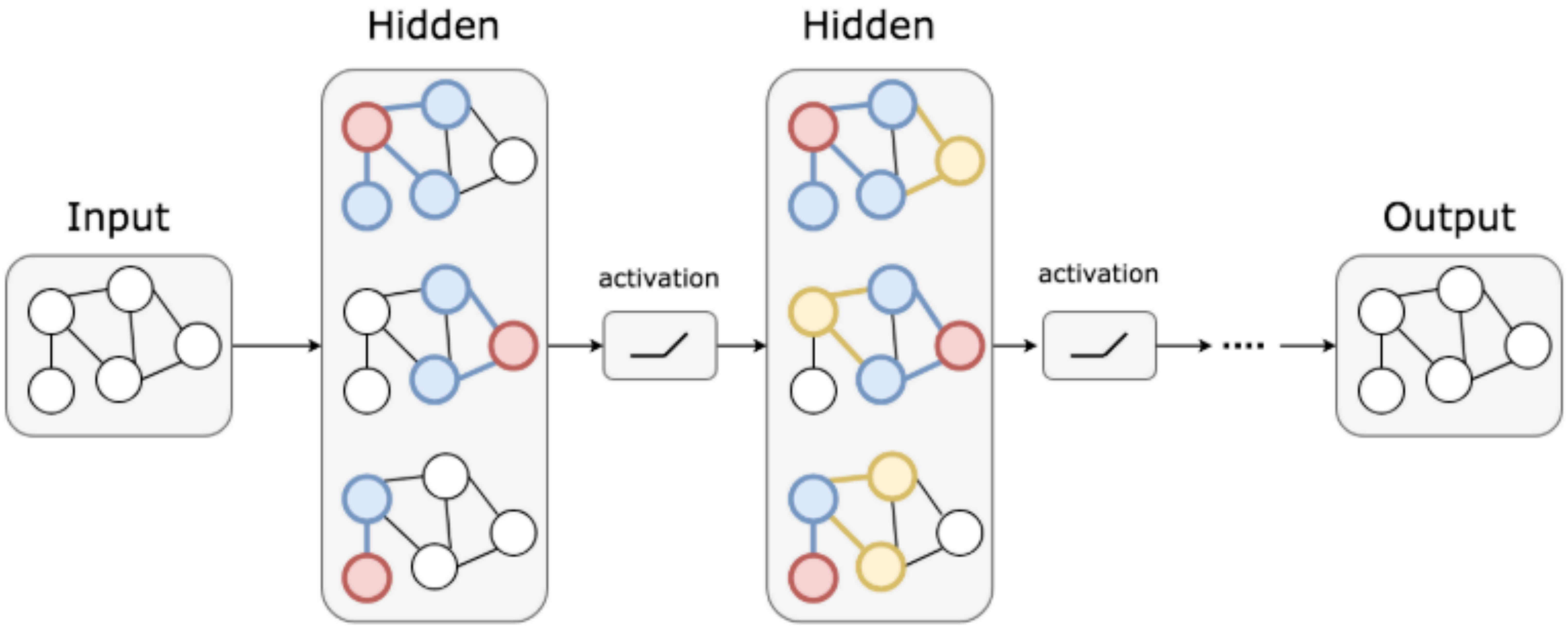
GCNs for ABSA
(Aspect-based Sentiment Analysis)

杨晰
xyang41@stu.ecnu.edu.cn

BGI: Graph Convolution Networks(GCN)

- 无向图G = (V, E)
  - V： 节点集合
  - E： 边集合



[ICLR2017]Semi-Supervised Classification with Graph Convolutional Networks

BGI: Graph Convolution Networks(GCN)

- 单层GCN

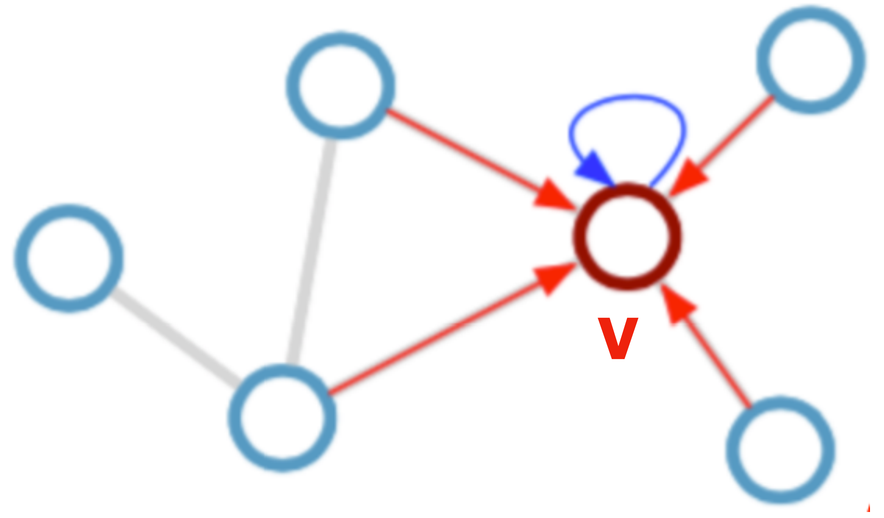
$$h_v = \boxed{ReLU}\left(\sum_{u\in\mathcal{N}(v)}\left(\boxed{W}\boxed{x_u}+\boxed{b}\right)\right)$$

激活函数

节点v的邻居节点集合

待学习参数

初始节点特征



- K层GCN

$$h_v^{(k+1)} = ReLU\left(\sum_{u\in\mathcal{N}(v)} W^{(k)}\boxed{h_u^{(k)}} + b^{(k)}\right)$$

[ICLR2017]Semi-Supervised Classification with Graph Convolutional Networks

BGII: Aspect-based Sentiment Analysis(ABSA)

- Classification Subtask: Given a sentence and a corresponding m-word aspect, predict the polarity (i.e. + / * / -)
- SemEval 2014 task 4 Dataset:

```
$T$ is super fast , around anywhere from 35 seconds to 1 minute .
Boot time
1
$T$ would not fix the problem unless I bought your plan for $ 150 plus .
tech support
-1
$T$ was easy .
Set up
1
Did not enjoy the new $T$ and touchscreen functions .
Windows 8
-1
Did not enjoy the new Windows 8 and $T$ .
touchscreen functions
-1
```

BGII: Aspect-based Sentiment Analysis(ABSA)

- Classification Subtask: Given a sentence and a corresponding m-word aspect, predict the polarity (i.e. + / * / -)
- Example:

Sentence Great food but the service was dreadful !

Inputs (Great food but the service was dreadful !, **food**) + **Outputs**

(Great food but the service was dreadful !, **service**) -

[EMNLP19] Aspect-based Sentiment Classification with Aspect-specific Graph Convolutional Networks

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

[1] *Beijing Institute of Technology,* [2] *University of Padua*

Motivations

- Attention-based models may focus on unrelated context words
- CNN-based models can only perceive consecutive-word features

Motivations

- Attention-based models may focus on unrelated context words
- CNN-based models can only perceive consecutive-word features

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

Motivations

- Attention-based models may focus on unrelated context words
- CNN-based models can only perceive consecutive-word features
- Syntactical dependency structures can capture long-range multi-word relations and syntactical information

Contributions

- First GCN-based model for aspect-based sentiment classification
- Propose an effective novel aspect-specific GCN model: ASGCN

ASGCN


Final Representation
(4) Aspect-aware Attention
(3) Aspect-specific Masking
(2) L-layer GCN
(1) BiLSTM
copy
Embeddings
Context

ASGCN: (2) GCN over DepTrees

- Conduct a position-aware transformation between GCN layers

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

$$q_i = \begin{cases} 1 - \frac{\tau+1-i}{n} & 1 \leq i < \tau + 1 \\ 0 & \tau + 1 \leq i \leq \tau + m \\ 1 - \frac{i-\tau-m}{n} & \tau + m < i \leq n \end{cases}$$

left context

aspect words

right context

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

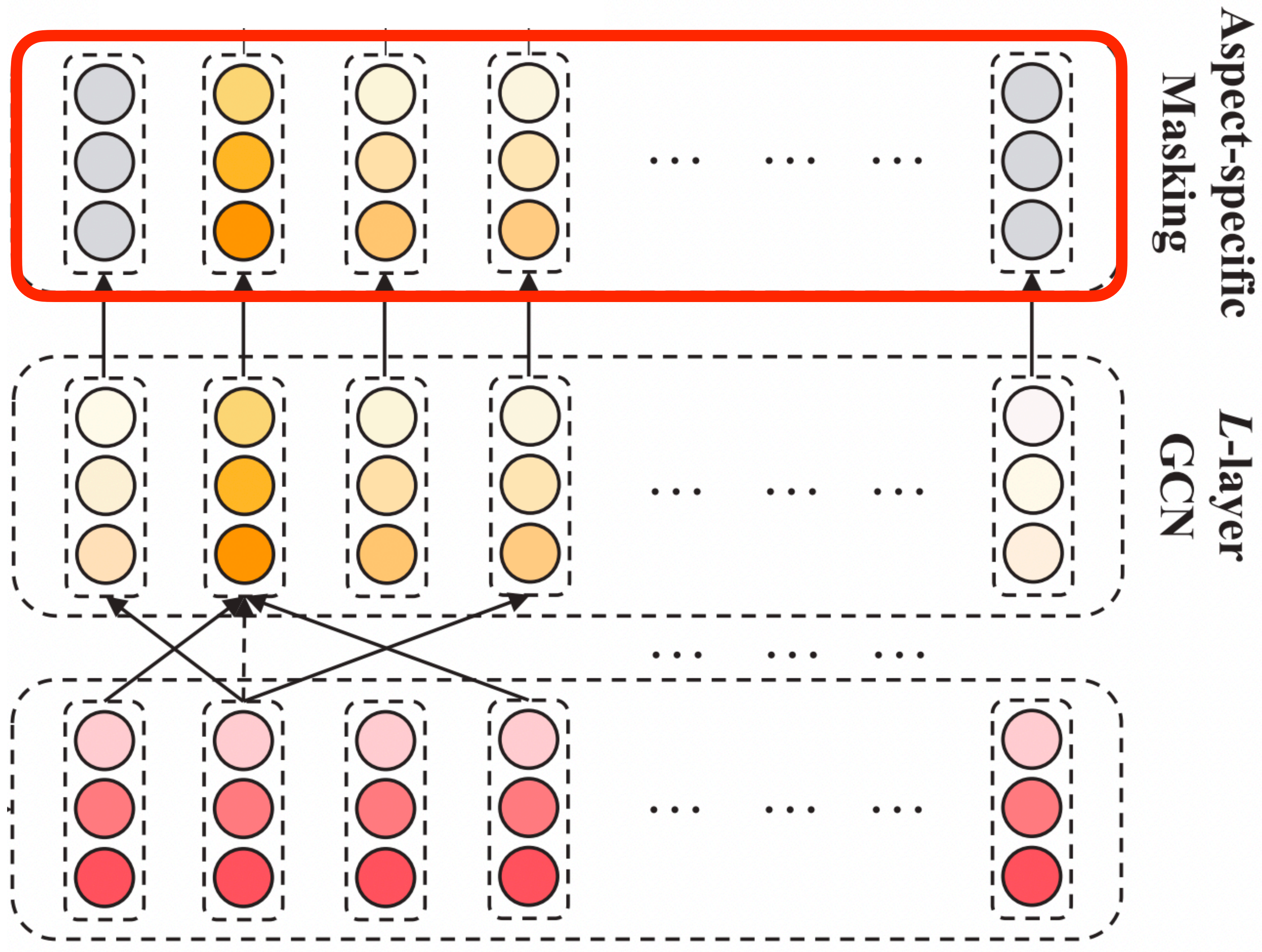
ASGCN: (3)(4)


Final Representation
(4) Aspect-aware Attention
(3) Aspect-specific Masking
(2) L-layer GCN
(1) BiLSTM
copy
Embeddings
Context

ASGCN: (3) Aspect-specific Masking

- Masking the context of given aspect

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



ASGCN: (4) Aspect-aware Attention

- $\boxed{\mathbf{h}_t^c}$: encoded by BiLSTM, $\boxed{\mathbf{h}_t^L}$: encoded by L-layers GCN
- Query = Aspect span, Key = Value = Contextual embeddings

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

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

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

Final Representation:

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

Training

$$\text{Loss} = -\sum_{(c,\hat{p})\in C} \log \mathbf{p}_{\hat{p}} + \lambda \|\Theta\|_2$$

Datasets

Dataset		# Pos.	# Neu.	# Neg.
TWITTER	Train	1561	3127	1560
	Test	173	346	173
LAP14	Train	994	464	870
	Test	341	169	128
REST14	Train	2164	637	807
	Test	728	196	196
REST15	Train	912	36	256
	Test	326	34	182
REST16	Train	1240	69	439
	Test	469	30	117

Experiments

- Main Results

Model	Twitter		Lap14		Rest14		Rest15		Rest16	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
SVM	63.40$^{\sharp}$	63.30$^{\sharp}$	70.49$^{\natural}$	N/A	80.16$^{\natural}$	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**†

DT： Directional

DG: Un-directional

Ablation Study

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.

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

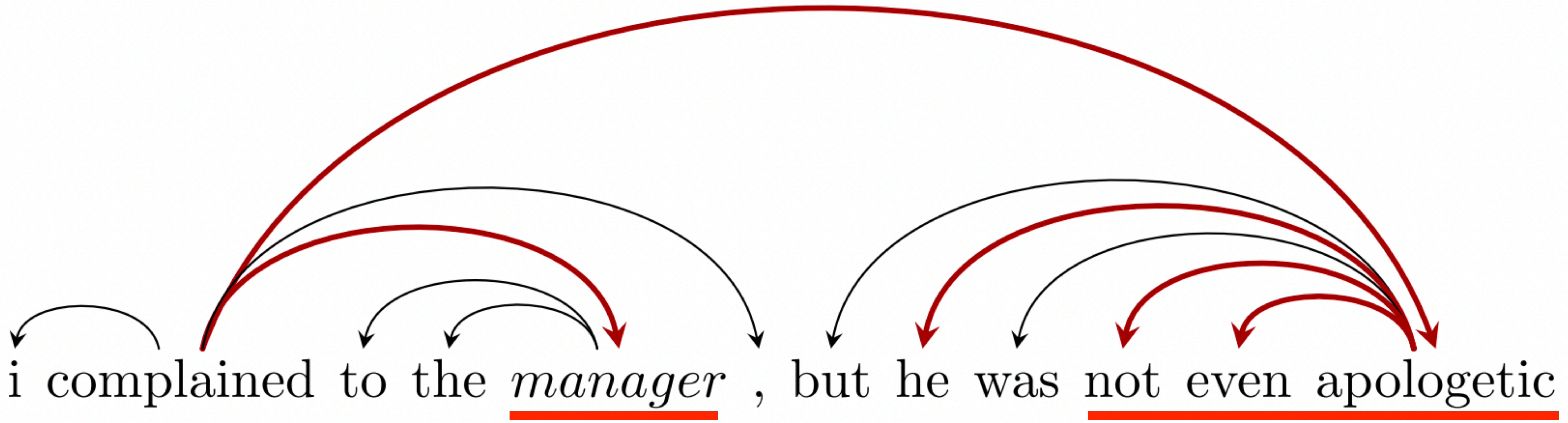
[EMNLP20] Inducing Target-Specific Latent Structures for Aspect Sentiment Classification

Chenhua Chen, Zhiyang Teng and Yue Zhang [1,2]
[1]*School of Engineering, Westlake University,*
[2]*Institute of Advanced Technology, Westlake Institute for Advanced Study*

Motivations

- Dependency parsing accuracies low on noisy texts (blogs and comments)
- Dependency syntax may not the most effective structure for ABSA

Motivations



(a) An example dependency tree from Stanford CoreNLP parser[2].

Motivations

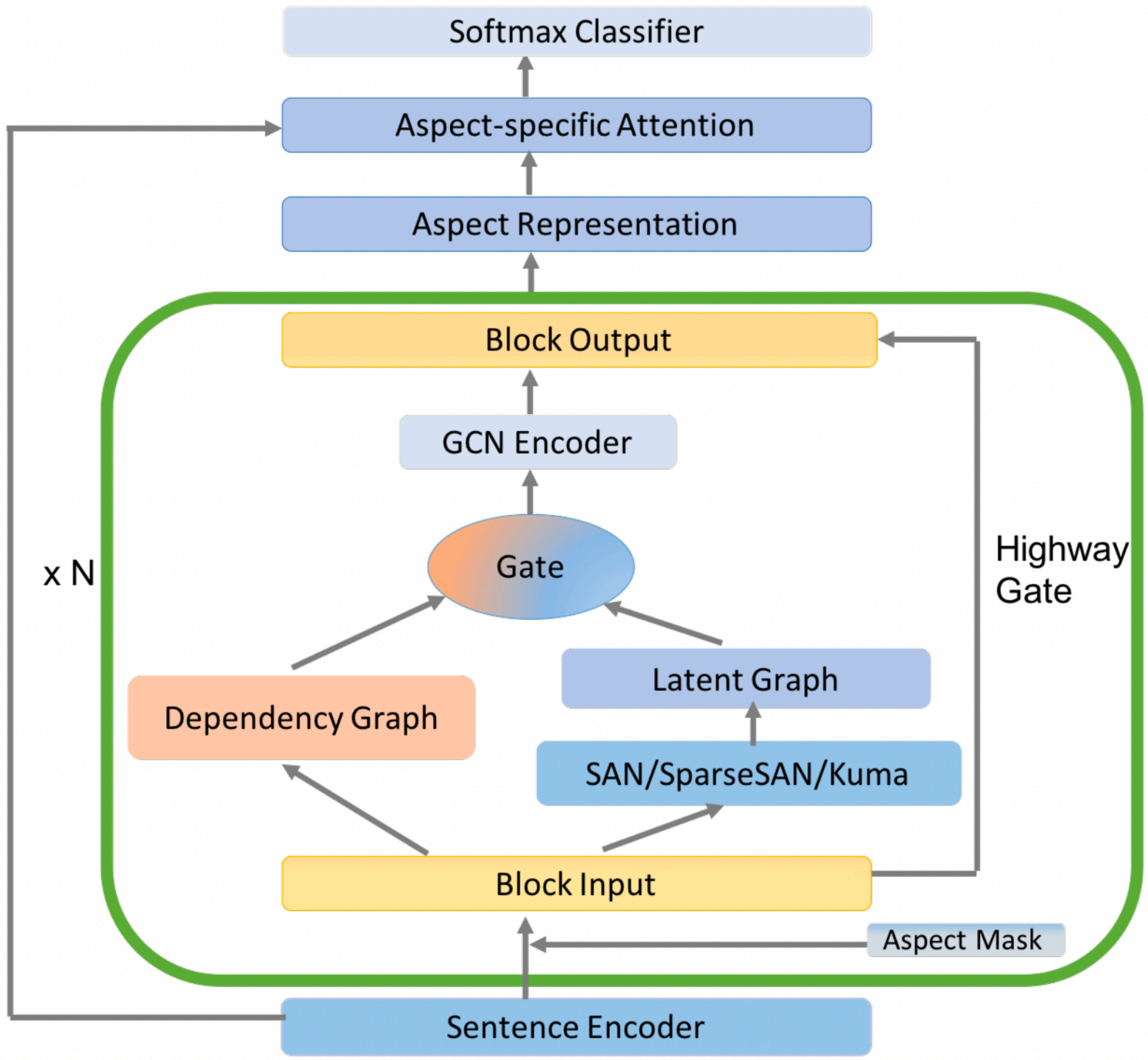
- Dependency parsing accuracies low on noisy texts (blogs and comments)
- Dependency syntax may not the most effective structure for ABSA

Automatically Induce Semantic Structures

Contributions

- First to investigate automatically inducing tree structures for ABSA
- SOTA

Model Architecture



Model Architecture: Sentence Encoder

- BiLSTM / BERT + Aspect-Mask

$$m_i = \begin{cases} 1 - \frac{f-i}{n} & 1 \le i < f, \\ 0 & f \le i \le e, \\ 1 - \frac{i-e}{n} & e < i \le n. \end{cases}$$

Model Architecture: Dependency Graph

$$\mathbf{A}_{dep}[i,j] = \begin{cases} 1 & \text{if } i \rightarrow j \text{ or } i \leftarrow j, \\ 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

Model Architecture: Latent Graph

- Self-attention-based latent graph (SANs)

$$\mathbf{A}_{lat} = \mathrm{softmax}\left(\frac{(\mathbf{Q}\mathbf{W}_q)(\mathbf{K}\mathbf{W}_k)^T}{\sqrt{d}}\right)$$

$$\mathbf{A}_{lat} = \frac{\sum_{i=1}^{K} \mathbf{A}^i_{head}}{K}$$

- Sparse-self-attentin-based latent graph

$$\mathbf{A}_{lat} = 1.5\text{-entmax}\left(\frac{(\mathbf{Q}\mathbf{W}_q)(\mathbf{K}\mathbf{W}_k)^T}{\sqrt{d}}\right)$$

- HardKuma-based latent graph: *produce stochastic graphs by sampling*

Model Architecture: Gated Combination

gated combine:

$$
\begin{aligned}
\mathbf{I}_{dep} &= \mathbf{A}_{dep}\mathbf{H}_{in}\mathbf{W}, \\
\mathbf{I}_{lat} &= \mathbf{A}_{lat}\mathbf{H}_{in}\mathbf{W}, \\
\mathbf{g} &= \sigma(\mathbf{I}_{lat}), \\
\mathbf{I}_{com} &= (1-\lambda\mathbf{g}) \odot \mathbf{I}_{dep} + \lambda\mathbf{g} \odot \mathbf{I}_{lat}, \\
\mathbf{H}_{out} &= \rho(\mathbf{I}_{com} + \mathbf{b}),
\end{aligned}
$$

block:

$$
\begin{aligned}
\mathbf{g}_l &= \sigma(\mathbf{H}^{l-1}), \\
\mathbf{H}^l_{com} &= gatedcombine(\mathbf{H}^{l-1}, \mathbf{A}_{dep}, \mathbf{A}_{lat}), \\
\mathbf{H}^l &= \mathbf{g}_l \odot \mathbf{H}^l_{com} + (1-\mathbf{g}_l) \odot \mathbf{H}^{l-1},
\end{aligned}
$$

Model Architecture: Aspect-specific attention

$$\gamma_t = \sum_{i=\boxed{f}}^{\boxed{e}} \mathbf{ce}_t^0 \mathbf{H}_i^N,$$

$$\boldsymbol{\alpha} = \mathrm{softmax}(\gamma),$$

$$\mathbf{z} = \boldsymbol{\alpha}\mathbf{C},$$

$$\mathbf{p} = \mathrm{softmax}(\mathbf{W}_o\mathbf{z} + \mathbf{b}_o),$$

Experiments

Model	depGCN	sanGCN	sparseGCN	kumaGCN		
				full	-latent	-dep
Acc.	88.99	88.64	89.29	**89.39**	89.12	89.23
F1	67.48	69.37	72.14	**73.19**	70.89	72.04

Table 2: Model performances on REST16.

Model	TWITTER		LAP14		REST14		REST15		REST16		AVERAGE	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
LSTM	69.56	67.70	69.28	63.09	78.13	67.47	77.37	55.17	86.80	63.88	76.23	63.46
MemNet	71.48	69.90	70.64	65.17	79.61	69.64	77.31	58.28	85.44	65.99	76.90	65.80
IAN	72.50	70.81	72.05	67.38	79.26	70.09	78.54	62.65	84.74	55.21	77.42	65.23
TNet-LF	**72.98**	**71.43**	74.61	70.14	80.42	71.03	78.47	59.47	89.07	70.43	79.11	68.50
depGCN	72.15	70.40	75.55	71.05	80.77	72.02	79.89	61.89	88.99	67.48	79.47	68.57
sparseGCN	72.64	71.02	75.91	71.89	81.30	72.68	80.57	65.52	89.29	72.14	79.94	70.65
kumaGCN	72.45	70.77	**76.12**	**72.42**	**81.43**	**73.64**	**80.69**	**65.99**	**89.39**	**73.19**	**80.02**	**71.20**

Table 3: Main results on five benchmark datasets: averaged accuracy (*Acc.*) and F1 score.

Experiments

Model	TWITTER		LAP14		REST14		REST15		REST16	
	Acc.	**F1**	**Acc.**	**F1**	**Acc.**	**F1**	**Acc.**	**F1**	**Acc.**	**F1**
AEN_BERT (Song et al., 2019)	75.14	74.15	76.96	73.67	84.29	77.22	-	-	-	-
RGAT+BERT (Wang et al., 2020)	76.15	74.88	78.21	74.07	**86.60**	**81.35**	-	-	-	-
BERT-SPC (Devlin et al., 2019)	73.41	72.38	80.56	77.20	84.55	75.74	83.03	63.92	90.75	74.00
depGCN + BERT	75.58	74.58	81.19	77.67	85.00	78.79	85.23	70.13	91.56	77.31
kumaGCN + BERT	**77.89**	**77.03**	**81.98**	**78.81**	86.43	80.30	**86.35**	**70.76**	**92.53**	**79.24**

Table 4: Main results on five benchmark datasets when BERT is used.

Experiments

Model\Target	REST14	REST15	REST16
BERT-SPC	49.46/43.54	44.10/39.69	45.45/33.67
depGCN+BERT	63.12/55.83	56.83/47.68	62.99/45.73
kumaGCN+BERT	**72.14/61.77**	**65.31/52.31**	**71.10/49.67**

Table 6: Results for transfer learning from Twitter to the three datasets.

Experiments: Attention Distance

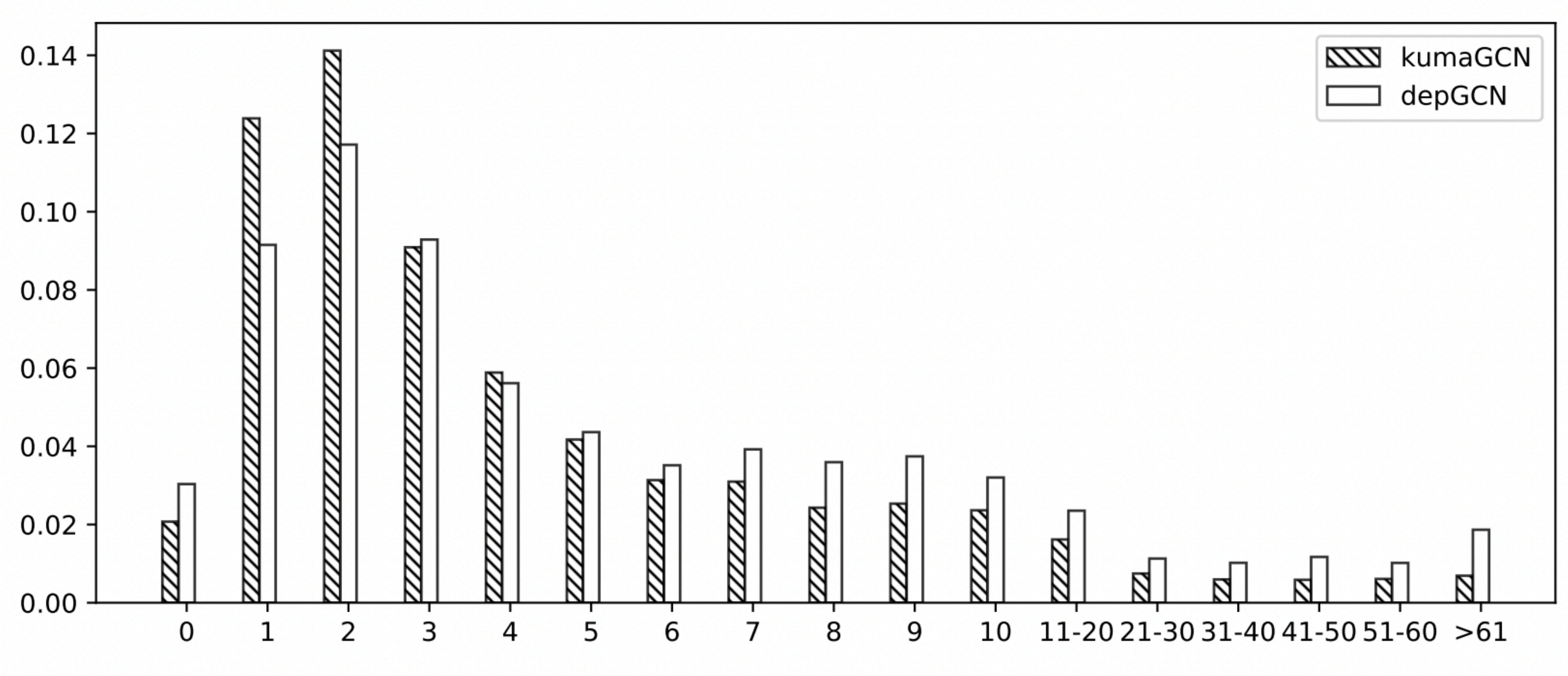


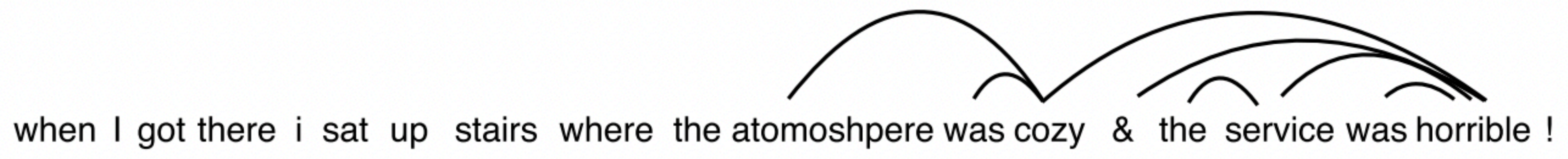
Figure 3: Attention distance distribution. x-axis: distance to the aspect terms; y-axis: attention scores.

Case Study

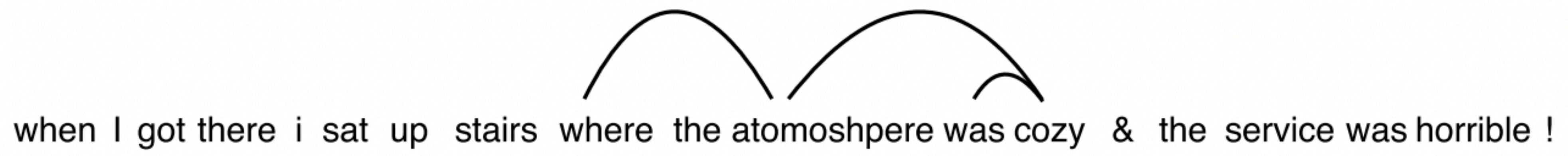
(depGCN) when$_{0.03}$ i$_{0.02}$ got$_{0.02}$ there$_{0.01}$ i $_{0.03}$ sat$_{0.05}$ up$_{0.03}$ stairs$_{0.04}$ where$_{0.09}$ the$_{0.07}$ *atmosphere*$_{0.01}$ was$_{0.02}$ cozy$_{0.03}$ &$_{0.07}$ the$_{0.07}$ service$_{0.02}$ was$_{0.11}$ horrible$_{0.25}$!$_{0.03}$

(kumaGCN) when$_{0.00}$ i$_{0.00}$ got$_{0.00}$ there$_{0.00}$ i$_{0.00}$ sat$_{0.00}$ up$_{0.00}$ stairs$_{0.00}$ where$_{0.09}$ the$_{0.10}$ *atmosphere*$_{0.14}$ was$_{0.31}$ cozy$_{0.35}$ &$_{0.00}$ the$_{0.00}$ service$_{0.00}$ was$_{0.00}$ horrible$_{0.00}$!$_{0.00}$

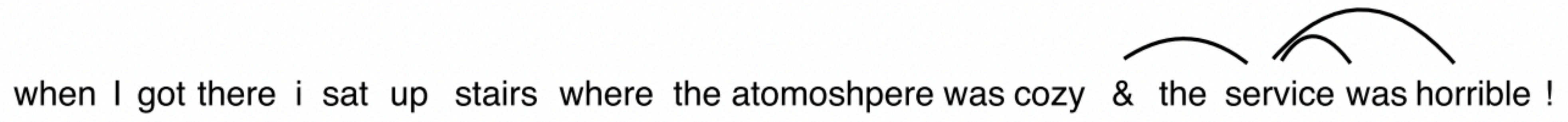
(a) Attention comparsions between depGCN and kumaGCN. Subscript numbers indicate the attention weights with respect to the underlined target words.



(b) Pruned dependency graph for "atmosphere" and "service".



(c) Latent graph for "atmosphere".



(d) Latent graph for "service".

Figure 4: Comparisons of graph representations.

Q & A