Recursive Neural Structural Correspondence Network for Cross-domain Aspect and Opinion Co-extraction

Wenya Wang[†‡] and Sinno Jialin Pan[†]

[†]Nanyang Technological University, Singapore

[‡]SAP Innovation Center Singapore

{wa0001ya, sinnopan}@ntu.edu.sg

July 18, 2018

Outline

Outline

Background: What is Aspect/Opinion Extraction

- **Fine-grained Opinion Mining**

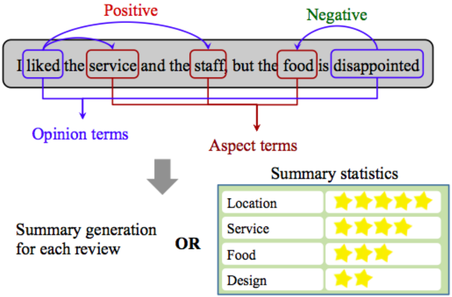


Figure 1: An example of review outputs.

  - **Our focus**: Aspect and Opinion Terms Co-extraction
  - **Challenge**: Limited resources for fine-grained annotations

Background: What is Aspect/Opinion Extraction

- **Fine-grained Opinion Mining**

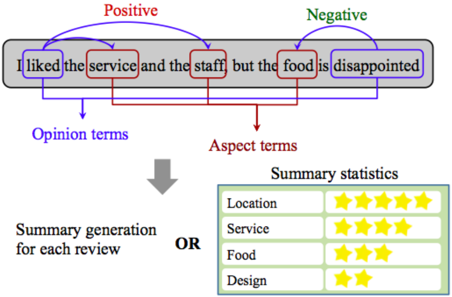


Figure 1: An example of review outputs.

- **Our focus**: Aspect and Opinion Terms Co-extraction
- **Challenge**: Limited resources for fine-grained annotations
  ⇒ Cross-domain extraction

Outline

Problem Definition

1. **Task formulation**: Sequence labeling

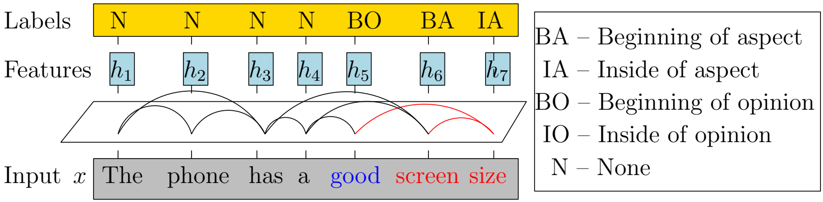


Figure 2: A deep learning model for sequence labeling.

2. **Domain Adaptation**
   - **Given**: Labeled data in source domain $\mathcal{D}_S = \{(\mathbf{x}_{S_i}, \mathbf{y}_{S_i})\}_{i=1}^{n_S}$, unlabeled data in target domain $\mathcal{D}_T = \{\mathbf{x}_{T_j}\}_{j=1}^{n_T}$
   - **Idea**: Build bridges across domains, learn shared space

Motivation: Domain Adaptation

1. **Domain shift & bridges**

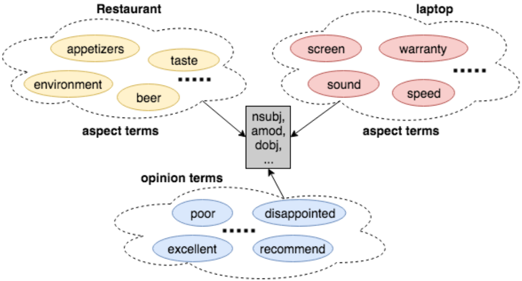


Figure 3: Domain shift for different domains.

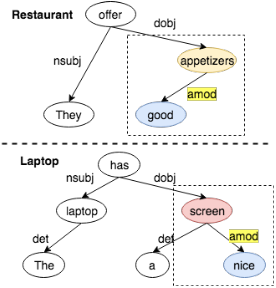


Figure 4: Syntactic patterns.

Motivation: Domain Adaptation

1. **Domain shift & bridges**

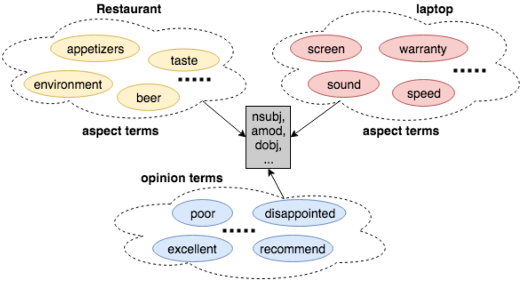


Figure 3: Domain shift for different domains.

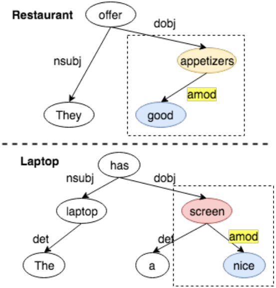


Figure 4: Syntactic patterns.

2. **Related work**
   - Adaptive bootstrapping [Li et al., 2012]
   - Auxiliary task with Recurrent neural network [Ding et al., 2017]

Outline

Overview & Contribution

- Recursive Neural Structural Correspondence Network (RNSCN)
  - Structural correspondences are built based on common syntactic structures
  - Use relation vectors with auxiliary labels to learn a shared space across domains
- Label denoising auto-encoder
  - Deal with auxiliary label noise
  - Group relation vectors into their intrinsic clusters in an unsupervised manner
- A joint deep model

Outline

Model Architecture: Recursive Neural Network

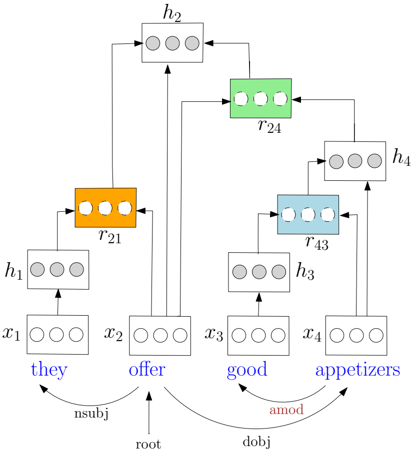


Figure 5: A recursive neural network.

Domain Adaptation

- **Relation vectors**: Relations as embeddings in the feature space

$$\mathbf{r}_{43} = \tanh(\mathbf{W}_h \mathbf{h}_3 + \mathbf{W}_x \mathbf{x}_4)$$

$$\mathbf{h}_4 = \tanh(\mathbf{W}_{\text{amod}} \mathbf{r}_{43} + \mathbf{W}_x \mathbf{x}_4 + \mathbf{b})$$

Model Architecture: Recursive Neural Network

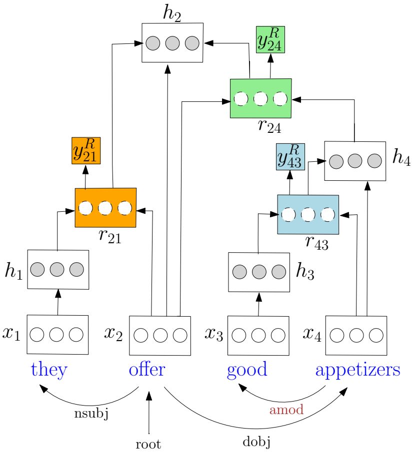


Figure 5: A recursive neural network.

Domain Adaptation

- **Relation vectors**: Relations as embeddings in the feature space

$$\mathbf{r}_{43} = \tanh(\mathbf{W}_h \mathbf{h}_3 + \mathbf{W}_x \mathbf{x}_4)$$
$$\mathbf{h}_4 = \tanh(\mathbf{W}_{amod} \mathbf{r}_{43} + \mathbf{W}_x \mathbf{x}_4 + \mathbf{b})$$

- **Auxiliary task**: Dependency relation prediction

$$\hat{\mathbf{y}}^R_{43} = \text{softmax}(\mathbf{W}_R \mathbf{r}_{43} + \mathbf{b}_R)$$

Model Architecture: Learn Shared Representations

Recursive **N**eural **S**tructural **C**orrespondence **N**etwork (RNSCN)

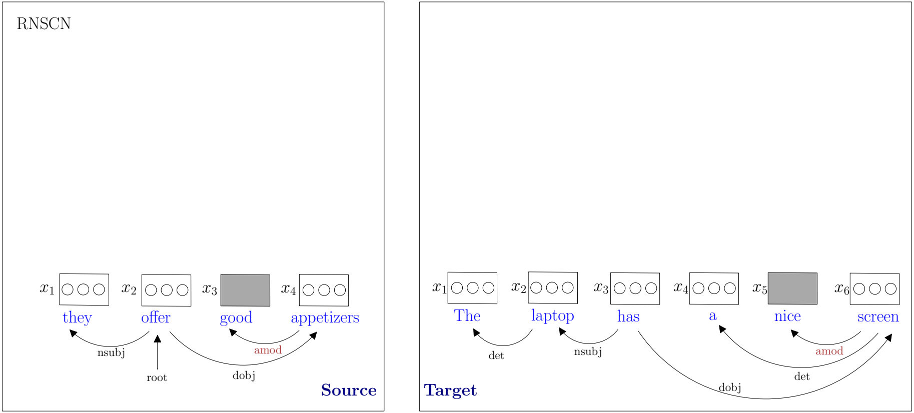


Figure 6: An example of how RNSCN learns the correspondences.

Model Architecture: Learn Shared Representations

Recursive **N**eural **S**tructural **C**orrespondence **N**etwork (RNSCN)

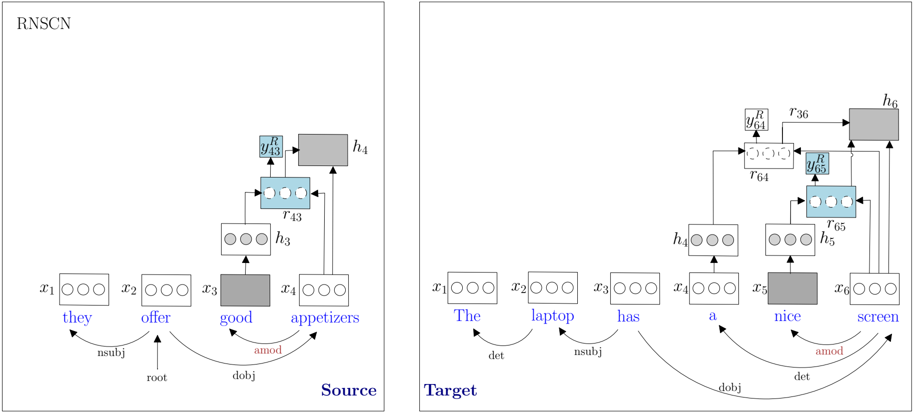
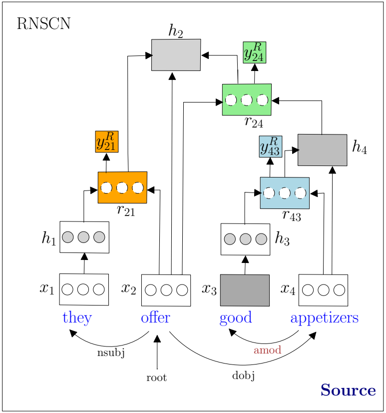


Figure 6: An example of how RNSCN learns the correspondences.

Model Architecture: Learn Shared Representations

Recursive **N**eural **S**tructural **C**orrespondence **N**etwork (RNSCN)



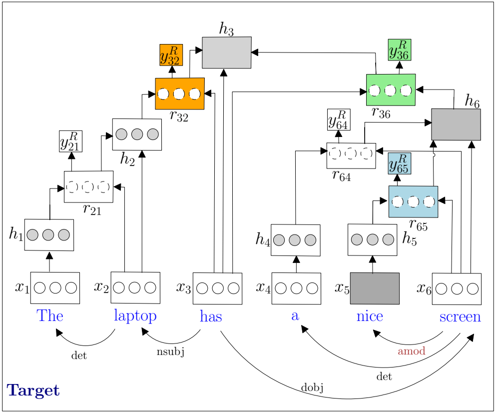


Figure 6: An example of how RNSCN learns the correspondences.

Model Architecture: Learn Shared Representations

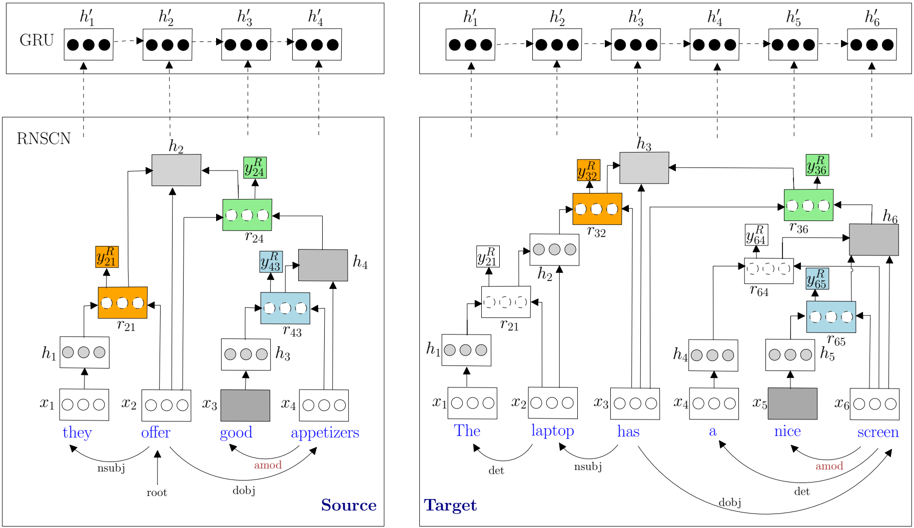


Figure 6: An example of how RNSCN learns the correspondences.

Model Architecture: Auxiliary Label Denoising

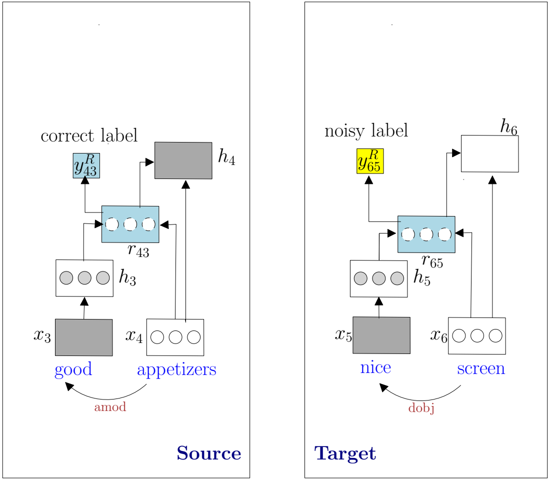


Figure 7: An autoencoder for label denoising.

Model Architecture: Auxiliary Label Denoising

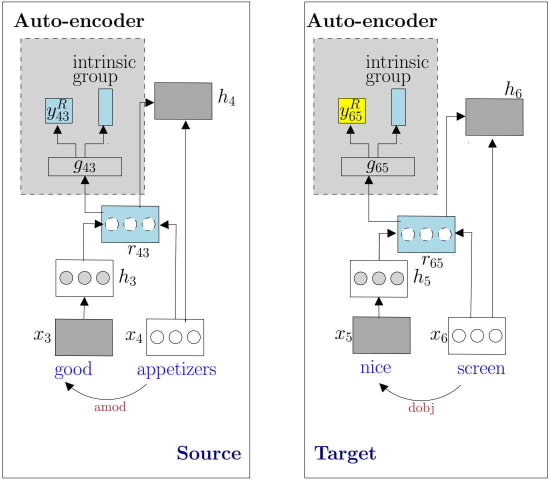


Figure 7: An autoencoder for label denoising.

Reduce label noise: auto-encoders

- Encoding:

$$\mathbf{g}_{nm} = f_{enc}(\mathbf{W}_{enc}, \mathbf{r}_{nm})$$

- Decoding:

$$\mathbf{r}'_{nm} = f_{dec}(\mathbf{W}_{dec}, \mathbf{g}_{nm})$$

- Auxiliary task:

$$\hat{\mathbf{y}}^R_{nm} = \text{softmax}(\mathbf{W}_R \mathbf{g}_{nm})$$

Model Architecture: Auxiliary Label Denoising

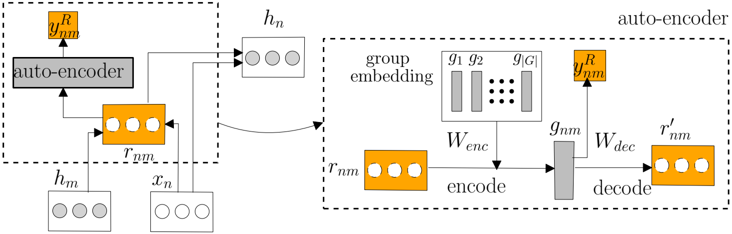


Figure 8: An autoencoder for relation grouping.

$$p(G_{nm} = i|\mathbf{r}_{nm}) = \frac{\exp(\mathbf{r}_{nm}^\top \mathbf{W}_{enc}\mathbf{g}_i)}{\sum_{j\in G}\exp(\mathbf{r}_{nm}^\top \mathbf{W}_{enc}\mathbf{g}_j)} \quad (1)$$

$$\mathbf{g}_{nm} = \sum_{i=1}^{|G|} p(G_{nm} = i|\mathbf{r}_{nm})\mathbf{g}_i \quad (2)$$

$$\ell_R = \ell_{R_1} + \alpha\ell_{R_2} + \beta\ell_{R_3} \quad (3)$$

$$\ell_{R_1} = \|\mathbf{r}_{nm} - \mathbf{W}_{dec}\mathbf{g}_{nm}\|_2^2$$

$$\ell_{R_2} = \sum_{k=1}^{K} -\mathbf{y}_{nm[k]}^R \log \hat{\mathbf{y}}_{nm[k]}^R$$

$$\ell_{R_3} = \left\|\mathbf{I} - \bar{\mathbf{G}}^\top\bar{\mathbf{G}}\right\|_F^2$$

Outline

Experiments

Dataset	Description	# Sentences	Training	Testing
R	Restaurant	5,841	4,381	1,460
L	Laptop	3,845	2,884	961
D	Device	3,836	2,877	959

Table 1: Data statistics with number of sentences.

cross-domain baselines: CrossCRF, RAP, Hier-Joint

single-domain baselines: RNCRF, RNGRU

Models	R→L		L→R		D→R		D→L	
	AS	OP	AS	OP	AS	OP	AS	OP
CrossCRF	19.72 (1.82)	59.20 (1.34)	28.19 (0.58)	65.52 (0.89)	6.59 (0.49)	39.38 (3.06)	24.22 (2.54)	46.67 (2.43)
RAP	25.92 (2.75)	62.72 (0.49)	46.90 (1.64)	67.98 (1.05)	45.44 (1.61)	60.67 (2.15)	28.22 (2.42)	59.79 (4.18)
Hier-Joint	33.66 (1.47)	- -	48.10 (1.45)	- -	47.97 (0.46)	- -	34.74 (2.27)	- -
RNCRF	24.26 (3.97)	60.86 (3.35)	40.88 (2.09)	66.50 (1.48)	34.59 (1.34)	63.89 (1.59)	40.59 (0.80)	60.17 (1.20)
RNGRU	24.23 (2.41)	60.65 (1.04)	39.78 (0.61)	62.99 (0.95)	38.15 (2.82)	64.21 (1.11)	39.44 (2.79)	60.85 (1.25)
RNSCN-GRU	37.77 (0.45)	62.35 (1.85)	**53.18** (0.75)	71.44 (0.97)	**49.62** (0.34)	69.42 (2.27)	45.92 (1.14)	63.85 (1.97)
RNSCN^{+}-GRU	**40.43** (0.96)	**65.85** (1.50)	52.91 (1.82)	**72.51** (1.03)	48.36 (1.14)	**73.75** (1.76)	**51.14** (1.68)	**71.18** (1.58)

Table 2: Comparisons with different baselines.

Experiments

- Injecting noise into syntactic relations

Models	R→L		R→D		L→R		L→D		D→R		D→L	
	AS	OP	AS	OP	AS	OP	AS	OP	AS	OP	AS	OP
RNSCN-GRU	37.77	62.35	33.02	57.54	**53.18**	71.44	35.65	60.02	**49.62**	69.42	45.92	63.85
RNSCN-GRU (r)	32.97	50.18	26.21	53.58	35.88	65.73	32.87	57.57	40.03	67.34	40.06	59.18
RNSCN^{+}-GRU	**40.43**	**65.85**	**35.10**	**60.17**	52.91	**72.51**	**40.42**	**61.15**	48.36	**73.75**	**51.14**	**71.18**
RNSCN^{+}-GRU (r)	39.27	59.41	33.42	57.24	45.79	69.96	38.21	59.12	45.36	72.84	50.45	68.05

Table 3: Effect of auto-encoders for auxiliary label denoising.

- Words grouping learned from auto-encoders

Group 1	this, the, their, my, here, it, I, our, not
Group 2	quality, jukebox, maitre-d, sauces, portions, volume, friend, noodles, calamari
Group 3	in, slightly, often, overall, regularly, since, back, much, ago
Group 4	handy, tastier, white, salty, right, vibrant, first, ok
Group 5	get, went, impressed, had, try, said, recommended, call, love
Group 6	is, are, feels, believes, seems, like, will, would

Table 4: Case studies on word clustering

Experiments

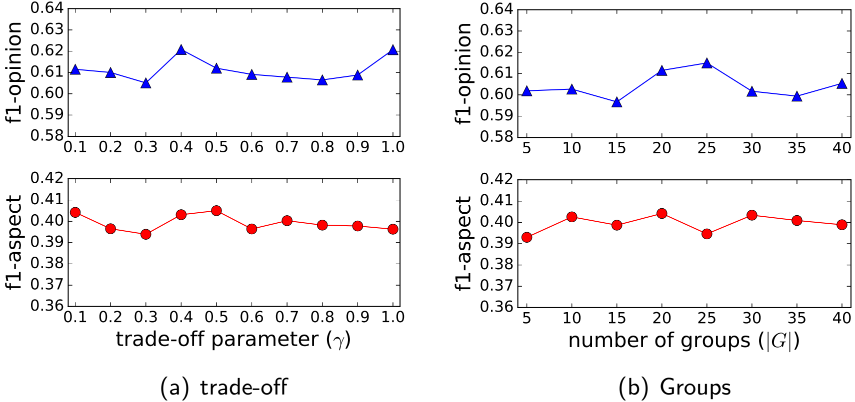


Figure 9: Sensitivity studies for L→D.

Domain Adaptation: Experiments

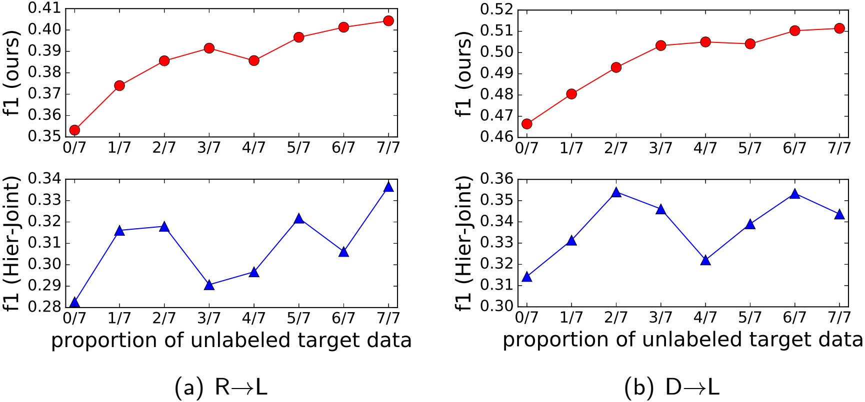


Figure 10: F1 vs proportion of unlabeled target data.

Outline

Conclusion

- A novel deep learning framework for Cross-domain aspect and opinion terms extraction.
- Embed syntactic structure into a deep model to bridge the gap between different domains.
- Apply auxiliary task to assist knowledge transfer.
- Address the problem of negative effect brought by label noise.
- Achieve promising results.

References

Ding, Y., Yu, J., and Jiang, J. (2017).
Recurrent neural networks with auxiliary labels for cross-domain opinion target extraction.
In *AAAI*.

Li, F., Pan, S. J., Jin, O., Yang, Q., and Zhu, X. (2012).
Cross-domain co-extraction of sentiment and topic lexicons.
In *ACL*.

Appendix: Domain Adaptation

Models	R→L		R→D		L→R		L→D		D→R		D→L	
	AS	OP	AS	OP	AS	OP	AS	OP	AS	OP	AS	OP
CrossCRF	19.72 (1.82)	59.20 (1.34)	21.07 (0.44)	52.05 (1.67)	28.19 (0.58)	65.52 (0.89)	29.96 (1.69)	56.17 (1.49)	6.59 (0.49)	39.38 (3.06)	24.22 (2.54)	46.67 (2.43)
RAP	25.92 (2.75)	62.72 (0.49)	22.63 (0.52)	54.44 (2.20)	46.90 (1.64)	67.98 (1.05)	34.54 (0.64)	54.25 (1.65)	45.44 (1.61)	60.67 (2.15)	28.22 (2.42)	59.79 (4.18)
Hier-Joint	33.66 (1.47)	- -	33.20 (0.52)	- -	48.10 (1.45)	- -	31.25 (0.49)	- -	47.97 (0.46)	- -	34.74 (2.27)	- -
RNCRF	24.26 (3.97)	60.86 (3.35)	24.31 (2.57)	51.28 (1.78)	40.88 (2.09)	66.50 (1.48)	31.52 (1.40)	55.85 (1.09)	34.59 (1.34)	63.89 (1.59)	40.59 (0.80)	60.17 (1.20)
RNGRU	24.23 (2.41)	60.65 (1.04)	20.49 (2.68)	52.28 (2.69)	39.78 (0.61)	62.99 (0.95)	32.51 (1.12)	52.24 (2.37)	38.15 (2.82)	64.21 (1.11)	39.44 (2.79)	60.85 (1.25)
RNSCN-CRF	35.26 (1.31)	61.67 (1.35)	32.00 (1.48)	52.81 (1.29)	53.38 (1.49)	67.60 (0.99)	34.63 (1.38)	56.22 (1.10)	48.13 (0.71)	65.06 (0.66)	46.71 (1.16)	61.88 (1.52)
RNSCN-GRU	37.77 (0.45)	62.35 (1.85)	33.02 (0.58)	57.54 (1.27)	53.18 (0.75)	71.44 (0.97)	35.65 (0.77)	60.02 (0.80)	49.62 (0.34)	69.42 (2.27)	45.92 (1.14)	63.85 (1.97)
RNSCN^h-GRU	39.13 (1.23)	63.65 (1.36)	33.97 (1.49)	59.24 (1.59)	**55.74** (2.27)	**75.20** (1.03)	40.30 (0.50)	60.57 (0.93)	**51.23** (0.42)	71.93 (1.55)	48.35 (1.00)	68.20 (1.11)
RNSCN$^+$-GRU	**40.43** (0.96)	**65.85** (1.50)	**35.10** (0.62)	**60.17** (0.75)	52.91 (1.82)	**72.51** (1.03)	**40.42** (0.70)	**61.15** (0.60)	48.36 (1.14)	**73.75** (1.76)	**51.14** (1.68)	**71.18** (1.58)

Table 5: Comparisons with different baselines.