Densely Connected Graph Convolutional Networks for Graph-to-Sequence Learning

Zhijiang Guo

Joint work with Yan Zhang, Zhiyang Teng, Wei Lu

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

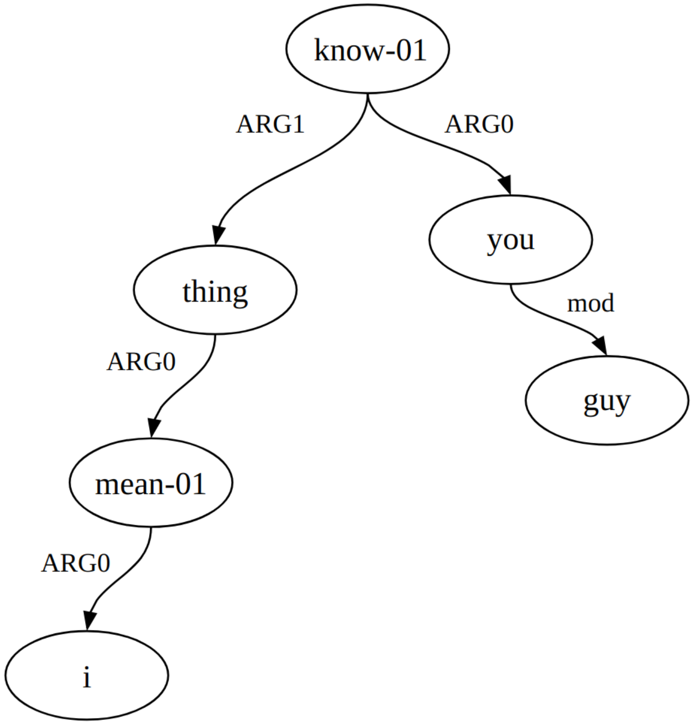
WIAS 浙江西湖高等研究院 WESTLAKE INSTITUTE FOR ADVANCED STUDY

Graph-to-Sequence Learning

AMR-to-Text Generation

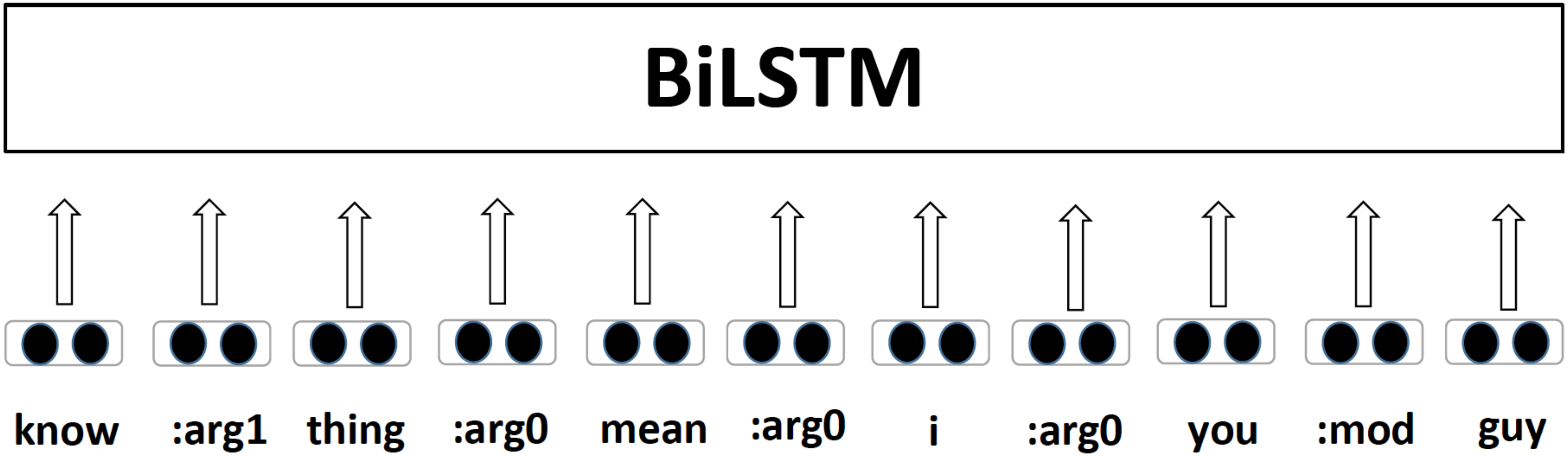
Syntax-Based Machine Translation

AMR-to-Text Generation



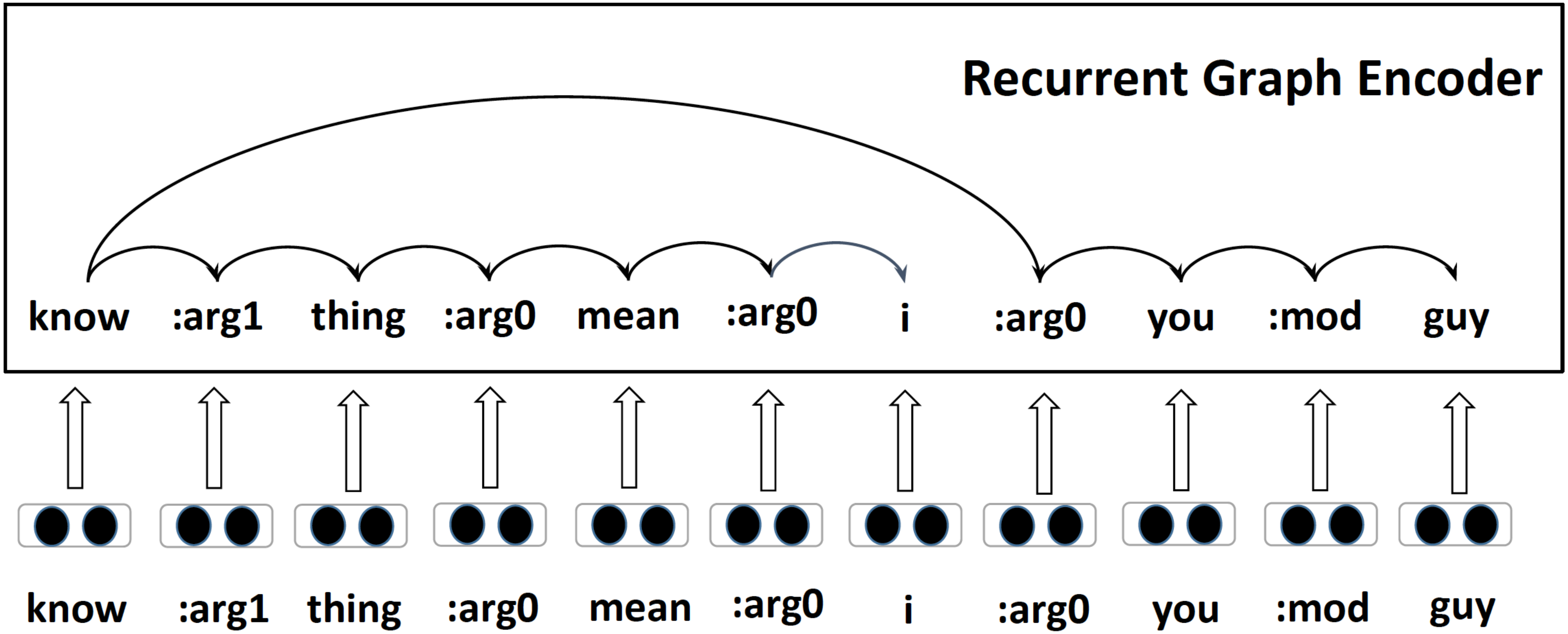
You guys know what I mean.

Sequence Encoder



(Konstas et al. 2017)

Recurrent Graph Encoder

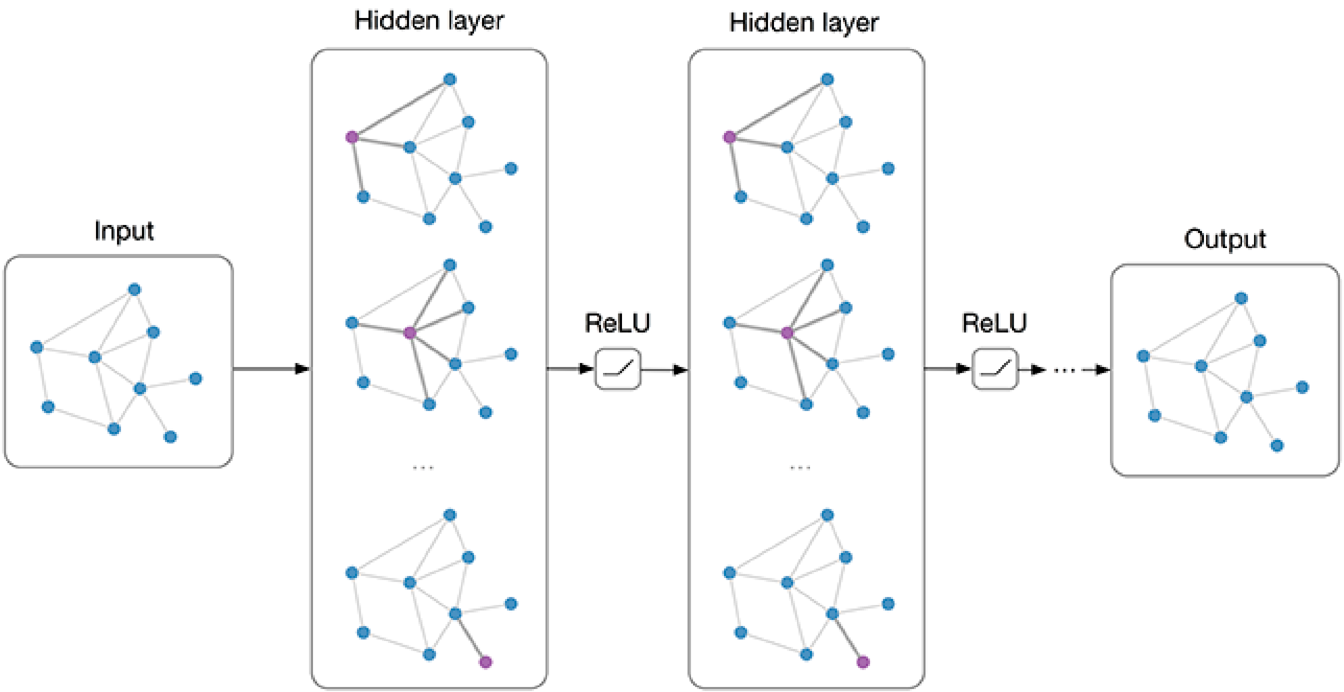


Graph State LSTM (Song et al., 2018)

Gated Graph Neural Networks (Beck et al., 2018)

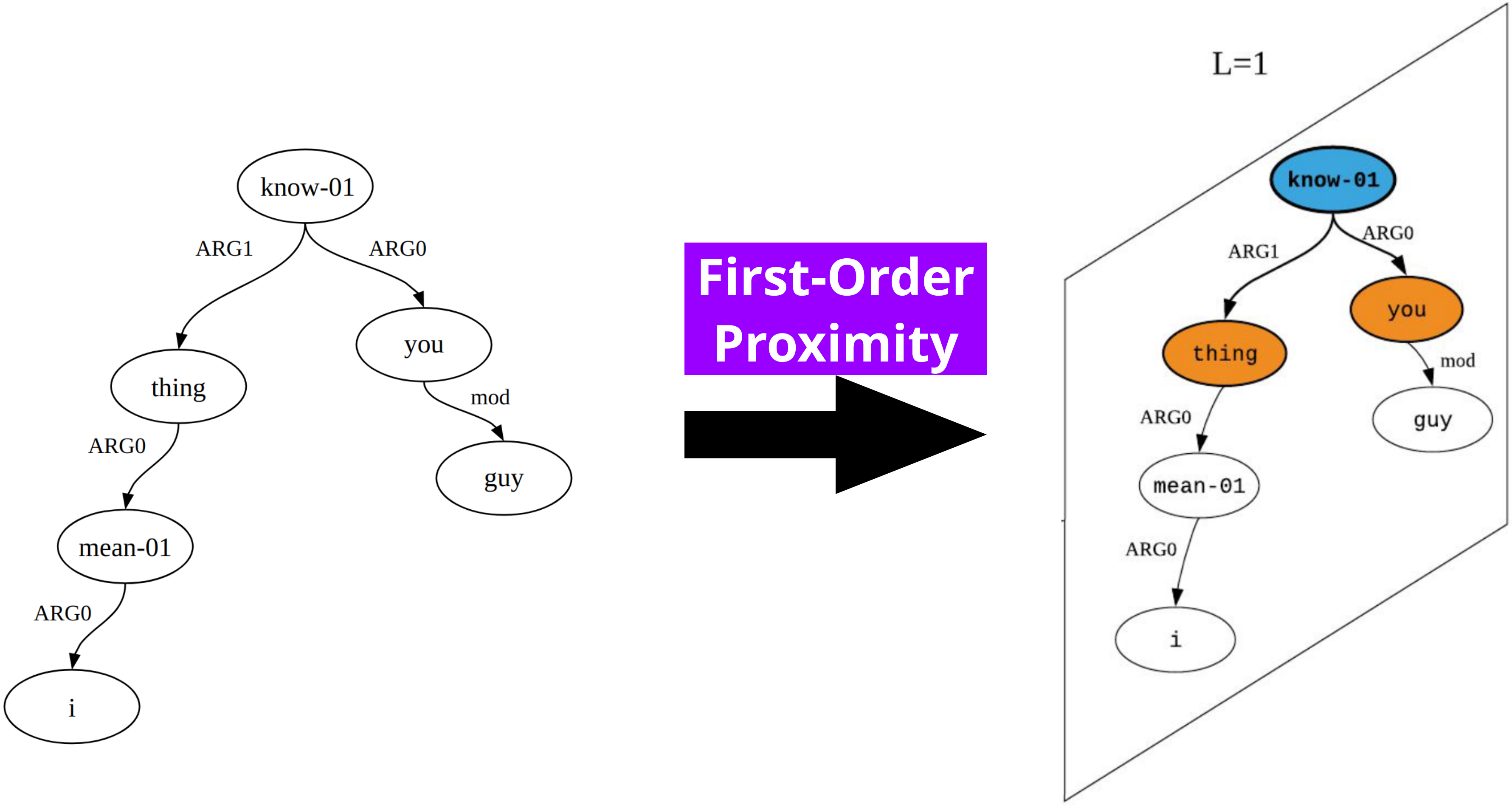
GCNs

Empirically, the best performance of GCNs is achieved with a **2-layer** model (Li et al., 2018, Xu et al., 2018)



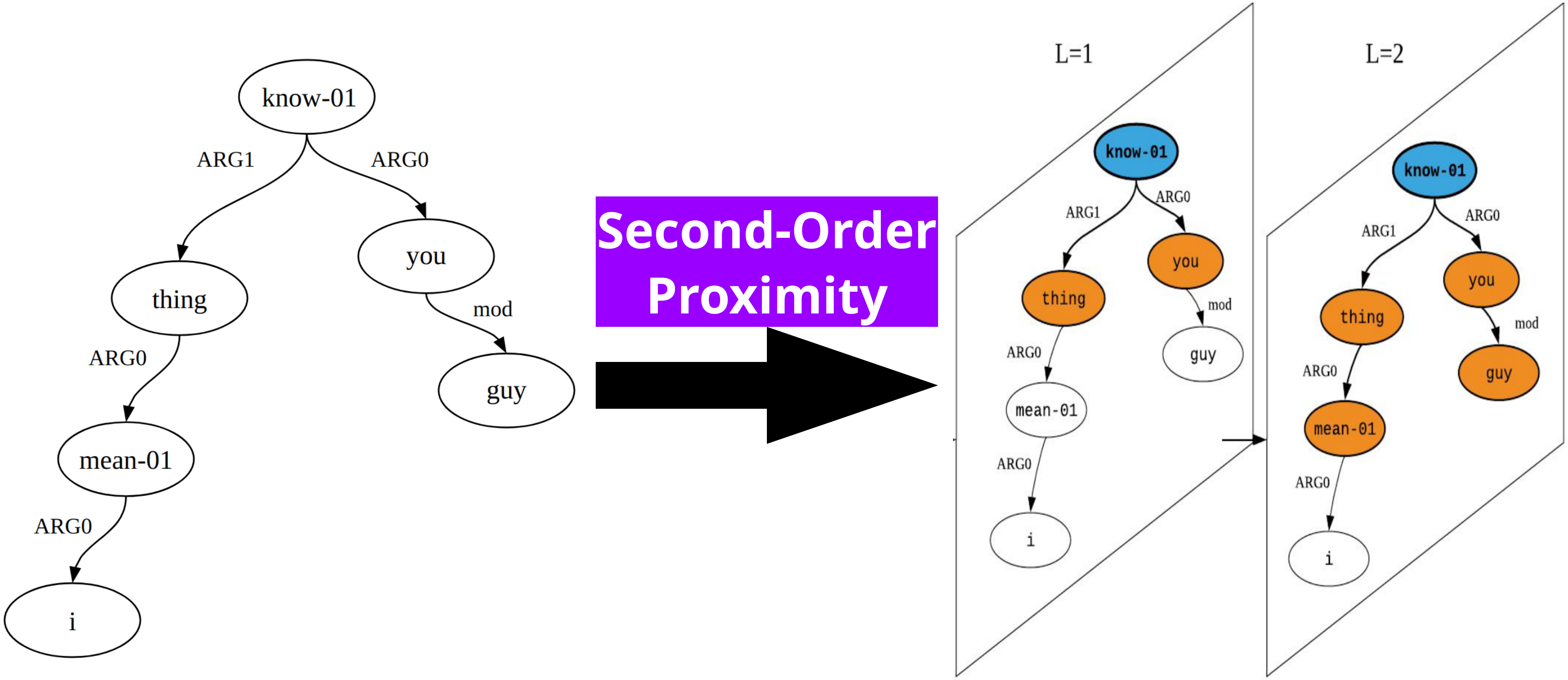
GCNs

first convolutional layer captures **first-order proximity** (immediate neighbors) information

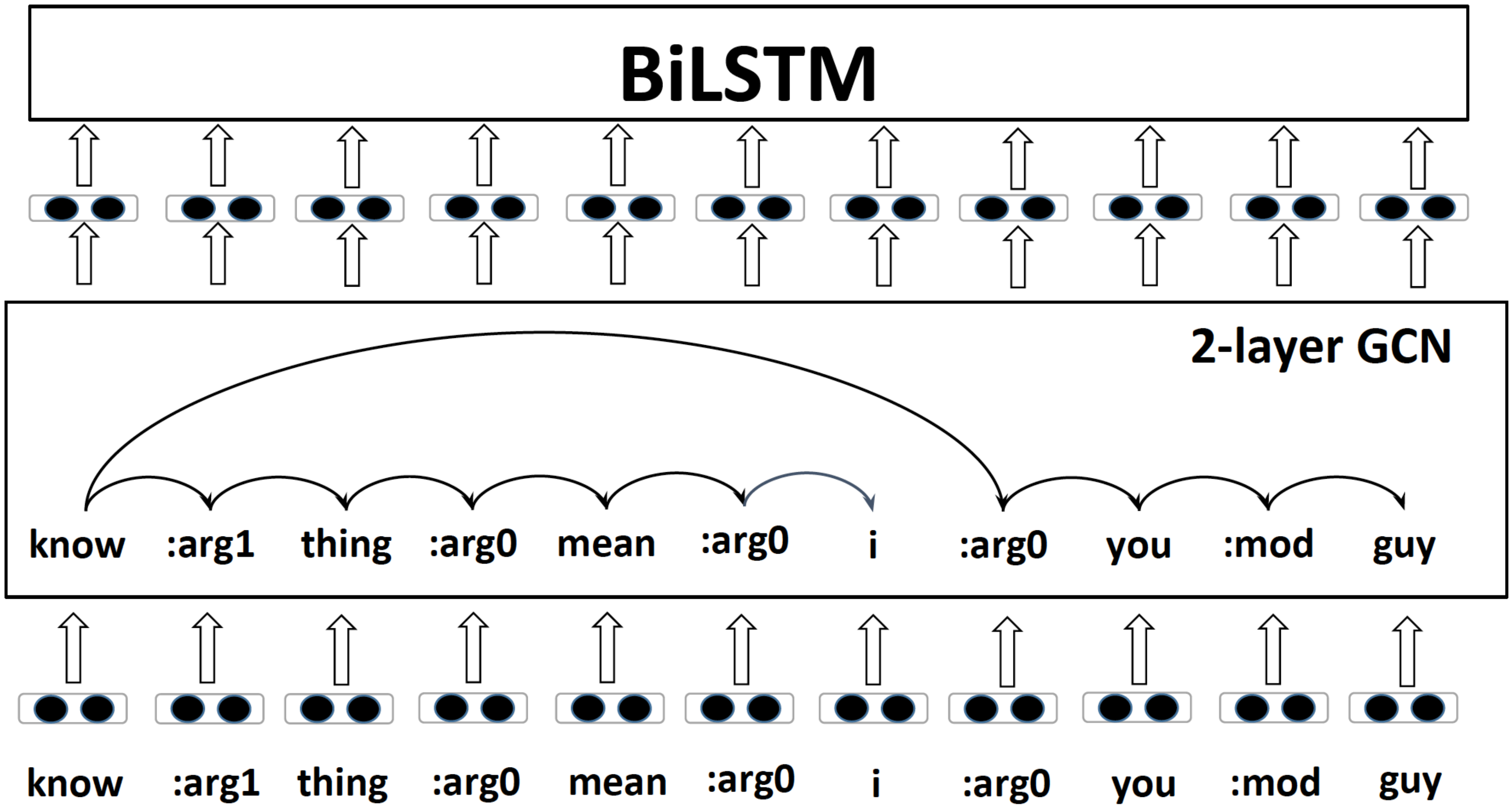


GCNs

second convolutional layer captures **second-order proximity** information



Convolutional Graph Encoder



(Bastings et al., 2017) (Damonte and Cohen , 2019)

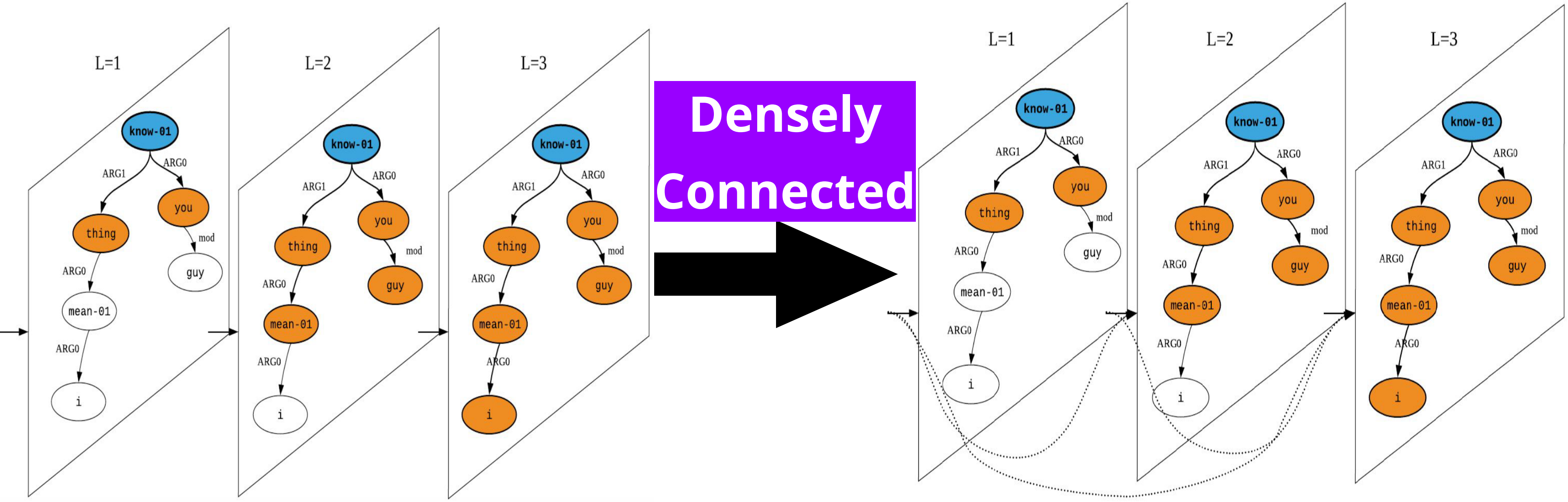
Motivation

Is it possible to build a more expressive GCN model to learn a better graph representation without relying on additional LSTM?

Densely Connected Graph Convolutional Networks (DCGCNs)

Dense Connectivity

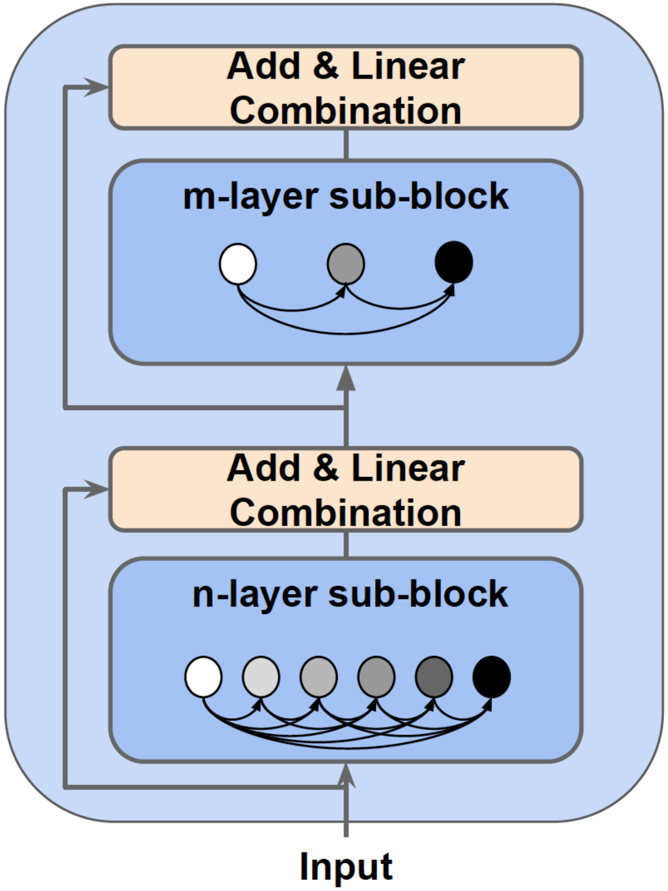
one layer takes inputs from **all preceding layers** rather than the previous layer only (Huang et al., 2017)



Densely Connected GCNs

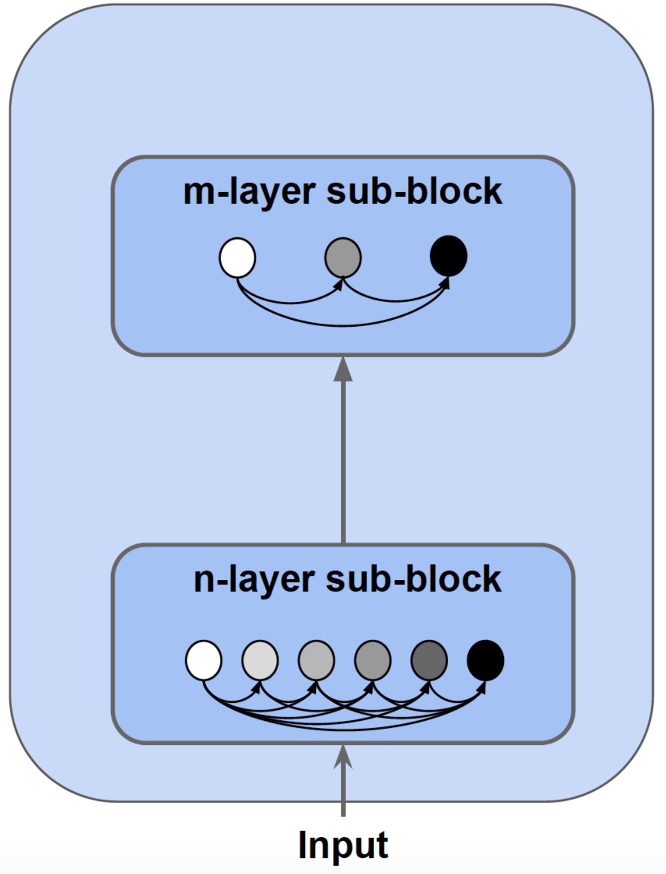
Stack Identical Blocks

- Densely Connected Sub-Block
- Linear Combination Layer



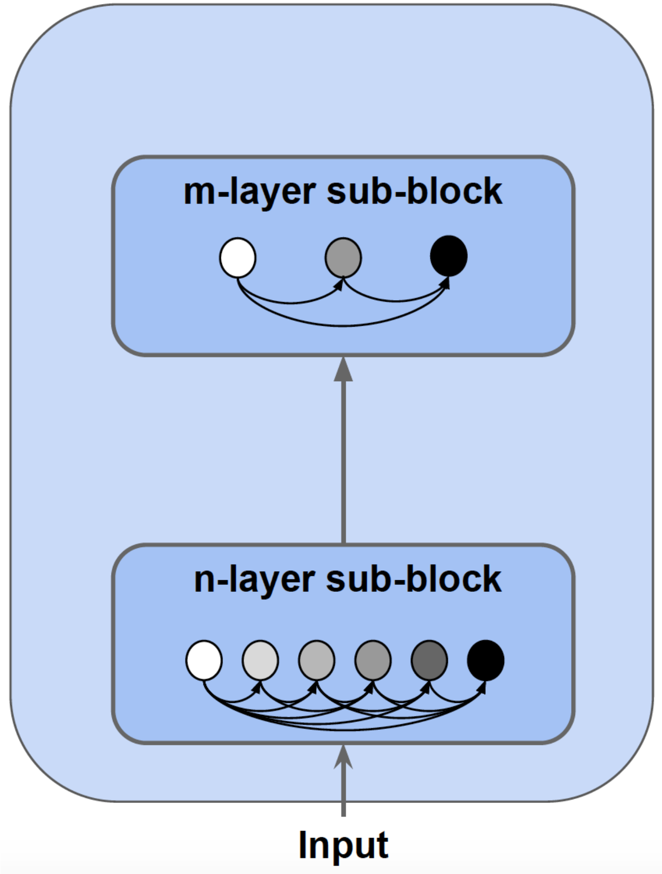
Densely Connected Sub-Block

Both sub-blocks are densely connected graph convolutional layers with different numbers (**m** and **n**) of layers.



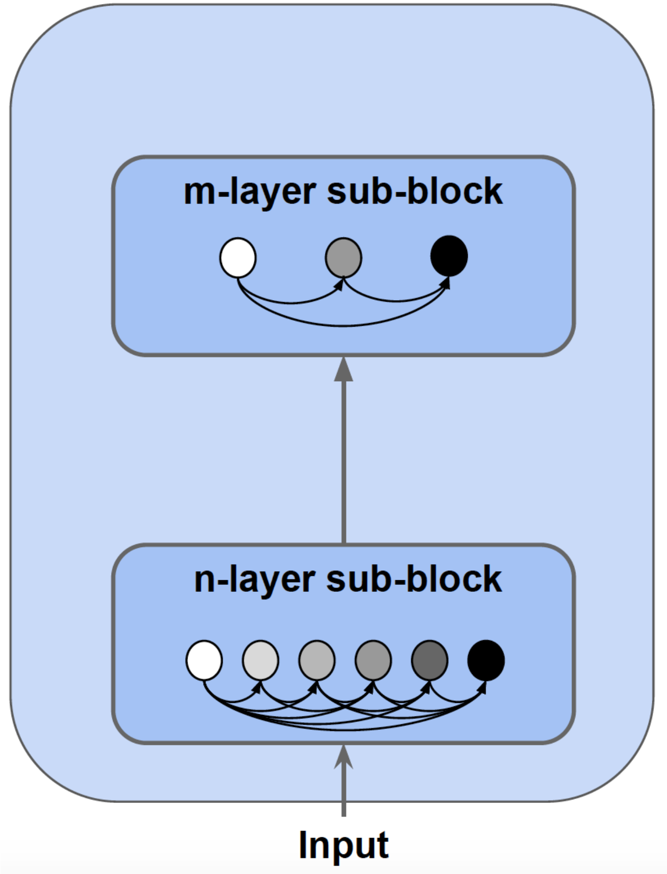
Densely Connected Sub-Block

Sub-blocks with different number of layers capture structural information at **different abstract levels**, similar to different filters.



Densely Connected Sub-Block

For **parameter efficiency**, the output dimension of each layer in the sub-block is designed to be **small.**



Densely Connected Sub-Block

Input dimension: 300

Sub-block layers: 3

Hidden dimension of each layer:

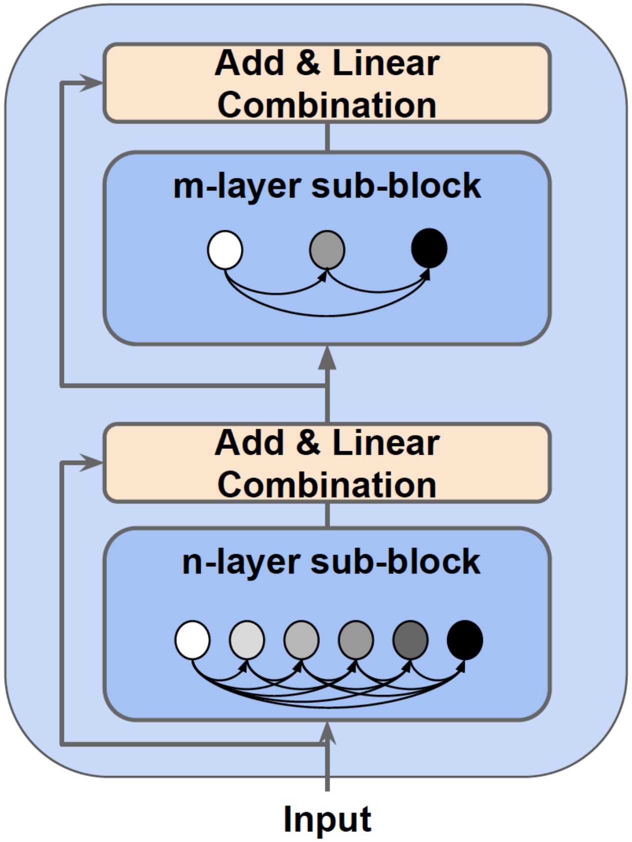
100 = 300 / 3 (proportional to #layers)

Output dimension: 300

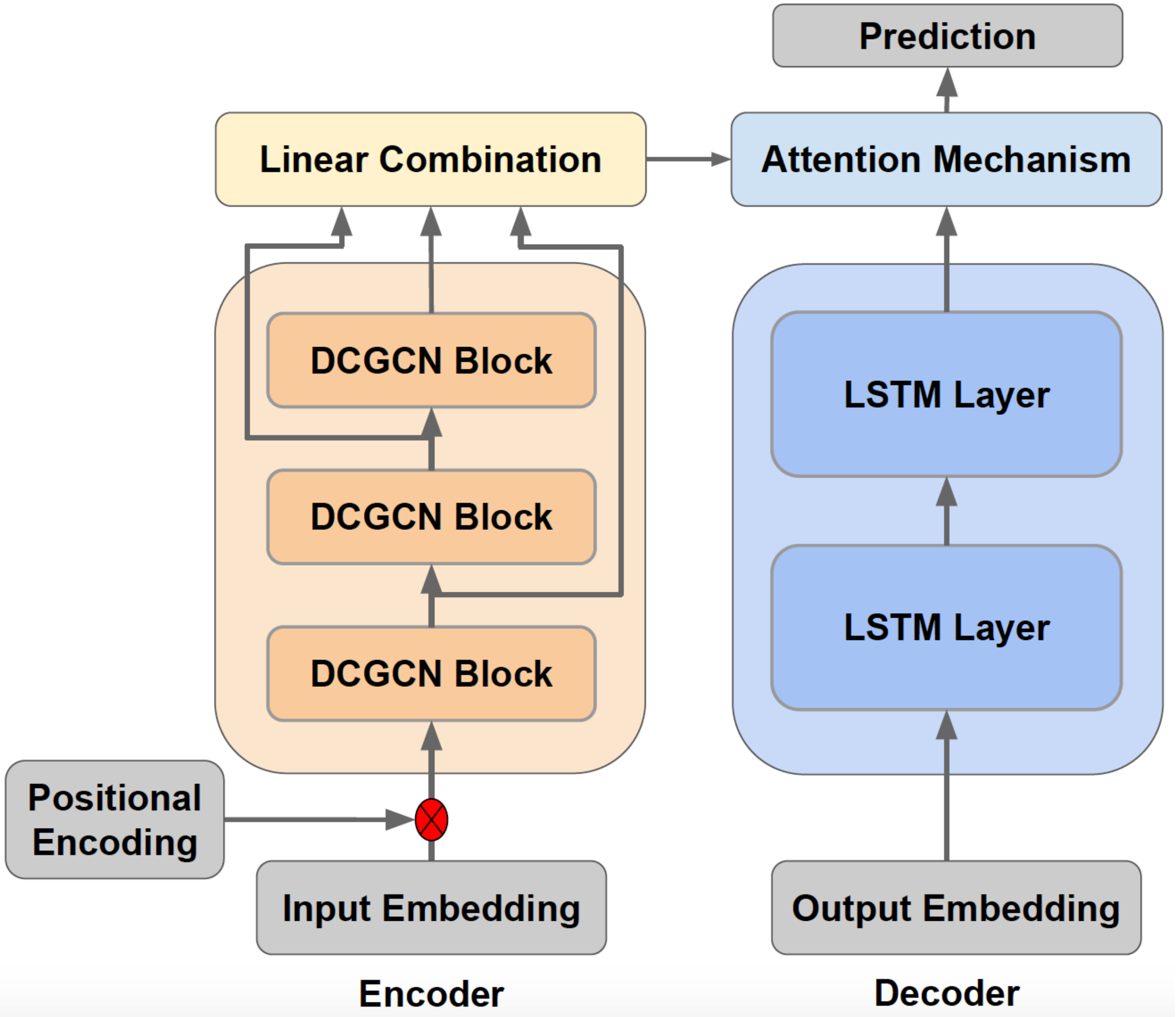
(concatenate output from all 3 layers)

Linear Combination Layer

This layer assigns **different weights** to outputs of different layers. Initial inputs of the sub-block are also incorporate by the residual connection.



Graph-to-Sequence Model



Experiments

AMR-to-Text Generation

- AMR 2015
- AMR 2017

Syntax-Based Machine Translation

- English-Czech (WMT 16)
- English-German (WMT 16)

Data Statistics

Dataset	Train	Dev	Test
AMR 2015	16,833	1,368	1,371
AMR 2017	36,521	1,368	1,371
En-Cs	181,112	2,656	2,999
En-De	226,822	2,169	2,999

AMR 2015

Sequential Encoder: LSTM (Konstas et al., 2017)

Graph Encoder: GS LSTM (Song et al., 2018)

GCN + LSTM (Damonte and Cohen , 2019)

Model	**External Data**	**BLEU**
LSTM	No	22.0
GS LSTM	No	23.3
GCN + LSTM	No	24.4
DCGCN	No	**25.7**

AMR 2015

Using External Training Data (0.2M)

Model	**External Data**	**BLEU**
LSTM	0.2M	27.4
GS LSTM	0.2M	28.2
DCGCN	0.1M	29.0
DCGCN	0.2M	**31.6**

AMR 2015

Using External Training Data (0.3M)

Model	**External Data**	**BLEU**
LSTM	2M	32.3
LSTM	20M	33.8
GS LSTM	2M	33.6
DCGCN (Single)	0.3M	33.2
DCGCN (Ensemble)	**0.3M**	**35.3**

AMR 2017 (Single)

Sequential Encoder: LSTM (Beck et al., 2017)

Graph Encoder: GGNNs (Beck et al., 2018)

GCN + LSTM (Damonte and Cohen , 2019)

Model	**#Parameters**	**BLEU**	**CHRF++**
LSTM	28.4M	21.7	49.1
GGNNs	28.3M	23.3	50.4
GCN + LSTM	N/A	24.5	N/A
DCGCN	**18.5M**	**27.6**	**57.3**

AMR 2017 (Ensemble)

Sequential Encoder: LSTM (Beck et al., 2017)

Graph Encoder: GGNNs (Beck et al., 2018)

Model	**#Parameters**	**BLEU**	**CHRF++**
LSTM	142.0M	26.6	52.5
GGNNs	141.0M	27.5	53.5
DCGCN	92.5M	**30.4**	**59.6**

English-German

Sequential Encoder: LSTM (Konstas et al., 2017)

Graph Encoder: GGNNs(Beck et al., 2018)

BoW/CNN/RNN + GCN (Bastings et al., 2017)

Model	Type	#Param	BLEU	CHRF++
BoW + GCN	Single	N/A	12.2	N/A
CNN + GCN	Single	N/A	13.7	N/A
BiRNN + GCN	Single	N/A	16.1	N/A
Seq2Seq	Single	41.4M	15.5	40.8
GGNNs	Single	41.2M	16.7	42.4
Our DCGCN	Single	**29.7M**	**19.0**	**44.1**

English-Gzech

Sequential Encoder: LSTM (Konstas et al., 2017)

Graph Encoder: GGNNs(Beck et al., 2018)

BoW/CNN/RNN + GCN (Bastings et al., 2017)

Model	**Type**	**#Param**	**BLEU**	**CHRF++**
BoW + GCN	Single	N/A	7.5	N/A
CNN + GCN	Single	N/A	8.7	N/A
BiRNN + GCN	Single	N/A	9.6	N/A
Seq2Seq	Single	41.4M	8.9	33.8
GGNNs	Single	41.2M	9.8	33.3
Our DCGCN	Single	**29.7M**	**12.1**	**37.1**

Density of Connection

Model	**BLEU**
DCGCN	25.5
- {4} dense block	24.8
- {3, 4} dense block	23.8
- {2, 3, 4} dense blocks	23.2

Ablation Test

Model	BLEU
DCGCN	25.5
- Global Node (GN)	24.2
- Linear Combination (LC)	23.7
- GN, LC	22.9

Conclusion

- DCGCNs allow the encoder to better capture the **rich structural** information of a graph, especially when it is large.

- Future: investigate how other NLP applications can potentially benefit from our proposed approach.

Thank You

Code available:

http://www.statnlp.org/research/machine-learning