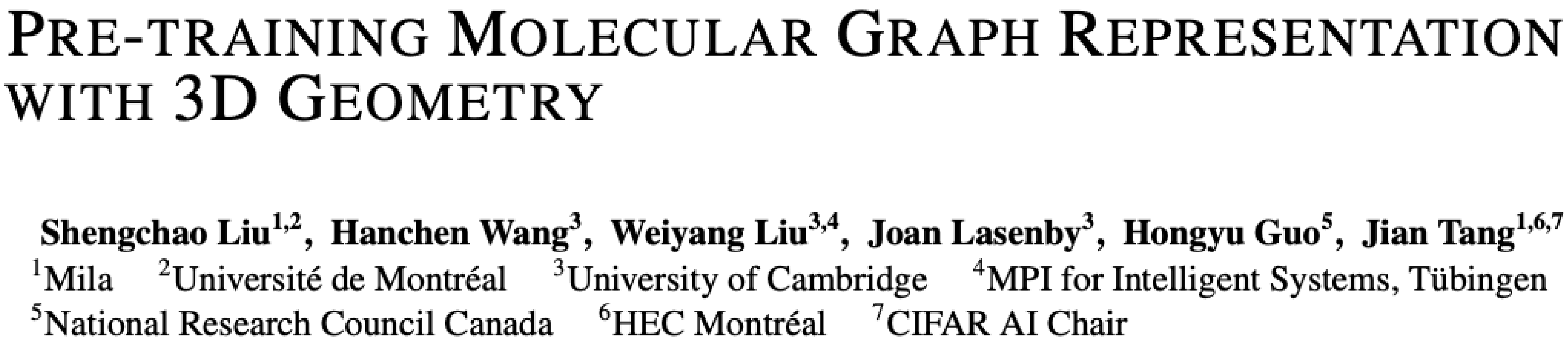


Pre-training Molecular Graph Representation with 3D Geometry

Shengchao Liu[1,2], **Hanchen Wang**[3], **Weiyang Liu**[3,4], **Joan Lasenby**[3], **Hongyu Guo**[5], **Jian Tang**[1,6,7]
[1]Mila [2]Université de Montréal [3]University of Cambridge [4]MPI for Intelligent Systems, Tübingen
[5]National Research Council Canada [6]HEC Montréal [7]CIFAR AI Chair

1. ***Introduction*** –

- Focuses on self-supervised molecular representation learning by leveraging the consistency between 2D topologies and **3D geometries**.

- As 3D info. can be scarce + tough to obtain, only required during pre-training.

- Two pre-training tasks, **contrastive-SSL + generative-SSL**.

- Most previous methods have focused on **2D topology**.

2. ***Brief Results*** –

- Proof-of-concept for 3D information.

- New **contrastive + generative SSL tasks**.

- Theoretical insights into why this works, maximize MI + privileged information.

3. <u>Preliminaries</u> –

- **GraphMVP** is based on a *view* design, i.e., each view is a *different* modality.

- **2D molecular graphs** normal, what we are used to. Notation –
  g_{2D} = (X, E) ;
  X is the atom attribute matrix,
  E is the bond attribute matrix.
  It's representation is h_{2D}, obtained using a GNN, after a *topological-transform*.
  h_{2D} = GNN-2D(T_{2D}(g_{2D}))

- **3D molecular graphs** includes spatial positions of atoms, in *continual motion on a potential energy surface.* Structures at local minima = **Conformers**. Notation –
  h_{3D} = GNN-3D(T_{3D}(g_{3D})) ;
  g_{3D} = (X, R) is *almost same* in representation, but R is the 3D-coordinate matrix,
  T_{3D} is a **3D-transformation**.

4. Method –

- Consider 2D vs 3D as two complimentary modalities.
- Pre-training → use both! Fine-tuning → **usually** only 2D is available.
- Pre-training is forcing the model to utilize **inter-data** and **intra-data** information to learn **local** and **global** distributions.
- The transformation functions T_{2D} and T_{3D} from below are just **masks**, mask nodes + corresponding edges.

- **Contrastive SSL between 2D and 3D**

$$\mathcal{L}_{\text{InfoNCE}} = -\frac{1}{2}\mathbb{E}_{p(\boldsymbol{x},\boldsymbol{y})}\Big[\log \frac{\exp(f_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{y}))}{\exp(f_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{y})) + \sum_j \exp(f_{\boldsymbol{x}}(\boldsymbol{x}^j,\boldsymbol{y}))}) + \log \frac{\exp(f_{\boldsymbol{y}}(\boldsymbol{y},\boldsymbol{x}))}{\exp(f_{\boldsymbol{y}}(\boldsymbol{y},\boldsymbol{x})) + \sum_j \exp(f_{\boldsymbol{y}}(\boldsymbol{y}^j,\boldsymbol{x}))}\Big]$$

$$\begin{aligned}\mathcal{L}_{\text{EBM-NCE}} = &-\frac{1}{2}\mathbb{E}_{p(\boldsymbol{y})}\Big[\mathbb{E}_{p_n(\boldsymbol{x}|\boldsymbol{y})}\log\big(1-\sigma(f_x(\boldsymbol{x},\boldsymbol{y}))\big) + \mathbb{E}_{p(\boldsymbol{x}|\boldsymbol{y})}\log\sigma(f_x(\boldsymbol{x},\boldsymbol{y}))\Big] \\ &-\frac{1}{2}\mathbb{E}_{p(\boldsymbol{x})}\Big[\mathbb{E}_{p_n(\boldsymbol{y}|\boldsymbol{x})}\log\big(1-\sigma(f_y(\boldsymbol{y},\boldsymbol{x}))\big) + \mathbb{E}_{p(\boldsymbol{y},\boldsymbol{x})}\log\sigma(f_y(\boldsymbol{y},\boldsymbol{x}))\Big]\end{aligned}$$

4. Method –

- **Generative SSL between 2D and 3D**

$$\log p(\boldsymbol{y}|\boldsymbol{x}) \geq \mathbb{E}_{q(\boldsymbol{z}_{\boldsymbol{x}}|\boldsymbol{x})}\big[\log p(\boldsymbol{y}|\boldsymbol{z}_{\boldsymbol{x}})\big] - KL(q(\boldsymbol{z}_{\boldsymbol{x}}|\boldsymbol{x})||p(\boldsymbol{z}_{\boldsymbol{x}}))$$

Clearly, first term is a **bottle-neck** ∵ graph data is **discrete**.

- So, model the **representation space** instead of the **data space**.

$$\begin{aligned}\mathcal{L}_{\text{G}} = \mathcal{L}_{\text{VRR}} = & \frac{1}{2}\Big[\mathbb{E}_{q(\boldsymbol{z}_{\boldsymbol{x}}|\boldsymbol{x})}\big[\|q_{\boldsymbol{x}}(\boldsymbol{z}_{\boldsymbol{x}}) - \textbf{SG}(h_{\boldsymbol{y}})\|^2\big] + \mathbb{E}_{q(\boldsymbol{z}_{\boldsymbol{y}}|\boldsymbol{y})}\big[\|q_{\boldsymbol{y}}(\boldsymbol{z}_{\boldsymbol{y}}) - \textbf{SG}(h_{\boldsymbol{x}})\|_2^2\big]\Big] \\ & + \frac{\beta}{2} \cdot \Big[KL(q(\boldsymbol{z}_{\boldsymbol{x}}|\boldsymbol{x})||p(\boldsymbol{z}_{\boldsymbol{x}})) + KL(q(\boldsymbol{z}_{\boldsymbol{y}}|\boldsymbol{y})||p(\boldsymbol{z}_{\boldsymbol{y}}))\Big].\end{aligned}$$

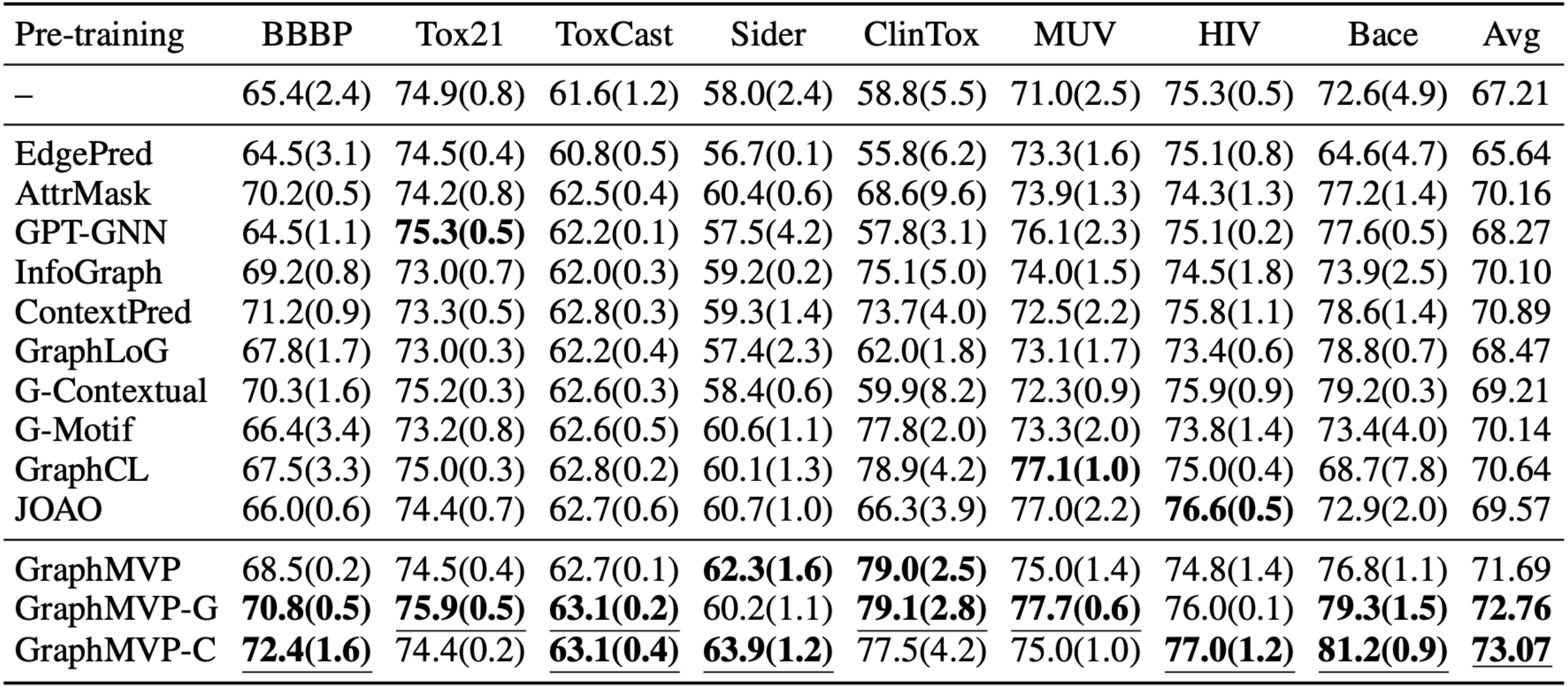
- **Multi-task objective** functions –

$$\mathcal{L}_{\text{GraphMVP}} = \alpha_1 \cdot \mathcal{L}_{\text{C}} + \alpha_2 \cdot \mathcal{L}_{\text{G}}$$

$$\mathcal{L}_{\text{GraphMVP-G}} = \mathcal{L}_{\text{GraphMVP}} + \alpha_3 \cdot \mathcal{L}_{\text{Generative 2D-SSL}}, \qquad \mathcal{L}_{\text{GraphMVP-C}} = \mathcal{L}_{\text{GraphMVP}} + \alpha_3 \cdot \mathcal{L}_{\text{Contrastive 2D-SSL}}$$

5. ***Experiments*** –

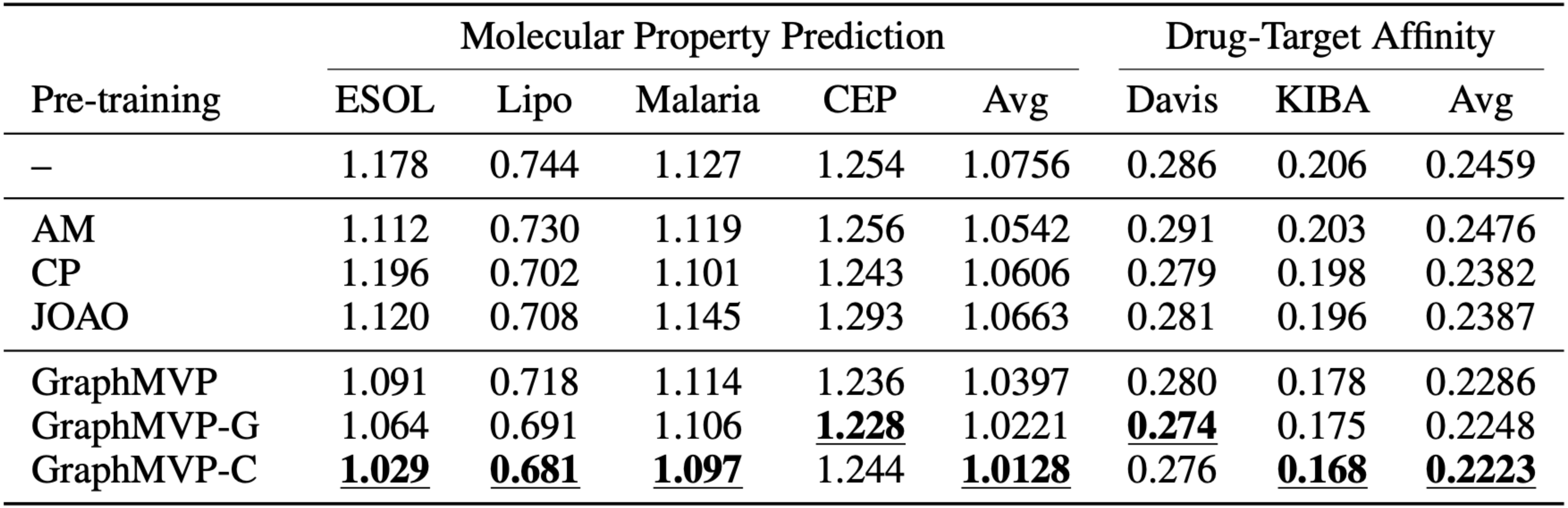
- Molecule property prediction



Pre-training	BBBP	Tox21	ToxCast	Sider	ClinTox	MUV	HIV	Bace	Avg
–	65.4(2.4)	74.9(0.8)	61.6(1.2)	58.0(2.4)	58.8(5.5)	71.0(2.5)	75.3(0.5)	72.6(4.9)	67.21
EdgePred	64.5(3.1)	74.5(0.4)	60.8(0.5)	56.7(0.1)	55.8(6.2)	73.3(1.6)	75.1(0.8)	64.6(4.7)	65.64
AttrMask	70.2(0.5)	74.2(0.8)	62.5(0.4)	60.4(0.6)	68.6(9.6)	73.9(1.3)	74.3(1.3)	77.2(1.4)	70.16
GPT-GNN	64.5(1.1)	**75.3(0.5)**	62.2(0.1)	57.5(4.2)	57.8(3.1)	76.1(2.3)	75.1(0.2)	77.6(0.5)	68.27
InfoGraph	69.2(0.8)	73.0(0.7)	62.0(0.3)	59.2(0.2)	75.1(5.0)	74.0(1.5)	74.5(1.8)	73.9(2.5)	70.10
ContextPred	71.2(0.9)	73.3(0.5)	62.8(0.3)	59.3(1.4)	73.7(4.0)	72.5(2.2)	75.8(1.1)	78.6(1.4)	70.89
GraphLoG	67.8(1.7)	73.0(0.3)	62.2(0.4)	57.4(2.3)	62.0(1.8)	73.1(1.7)	73.4(0.6)	78.8(0.7)	68.47
G-Contextual	70.3(1.6)	75.2(0.3)	62.6(0.3)	58.4(0.6)	59.9(8.2)	72.3(0.9)	75.9(0.9)	79.2(0.3)	69.21
G-Motif	66.4(3.4)	73.2(0.8)	62.6(0.5)	60.6(1.1)	77.8(2.0)	73.3(2.0)	73.8(1.4)	73.4(4.0)	70.14
GraphCL	67.5(3.3)	75.0(0.3)	62.8(0.2)	60.1(1.3)	78.9(4.2)	**77.1(1.0)**	75.0(0.4)	68.7(7.8)	70.64
JOAO	66.0(0.6)	74.4(0.7)	62.7(0.6)	60.7(1.0)	66.3(3.9)	77.0(2.2)	**76.6(0.5)**	72.9(2.0)	69.57
GraphMVP	68.5(0.2)	74.5(0.4)	62.7(0.1)	**62.3(1.6)**	**79.0(2.5)**	75.0(1.4)	74.8(1.4)	76.8(1.1)	71.69
GraphMVP-G	**70.8(0.5)**	**75.9(0.5)**	**63.1(0.2)**	60.2(1.1)	**79.1(2.8)**	**77.7(0.6)**	76.0(0.1)	**79.3(1.5)**	**72.76**
GraphMVP-C	**72.4(1.6)**	74.4(0.2)	**63.1(0.4)**	**63.9(1.2)**	77.5(4.2)	75.0(1.0)	**77.0(1.2)**	**81.2(0.9)**	**73.07**

5. *Experiments* –

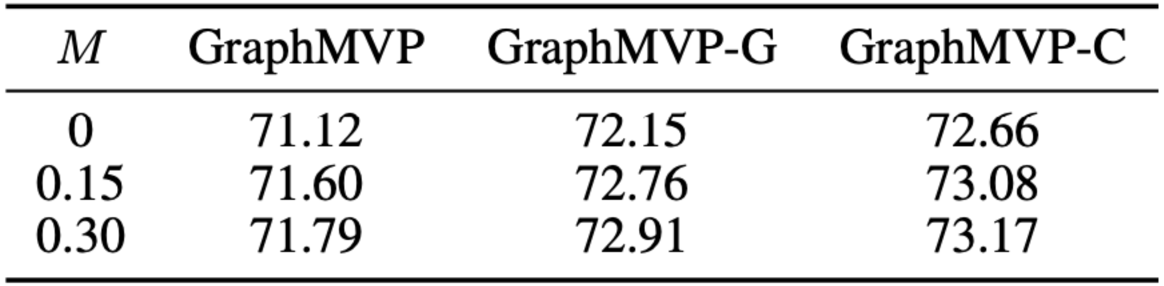
- Extra property prediction, using regression + also for Drug Affinity tasks. (**lower is better**, std is not reported because very small)



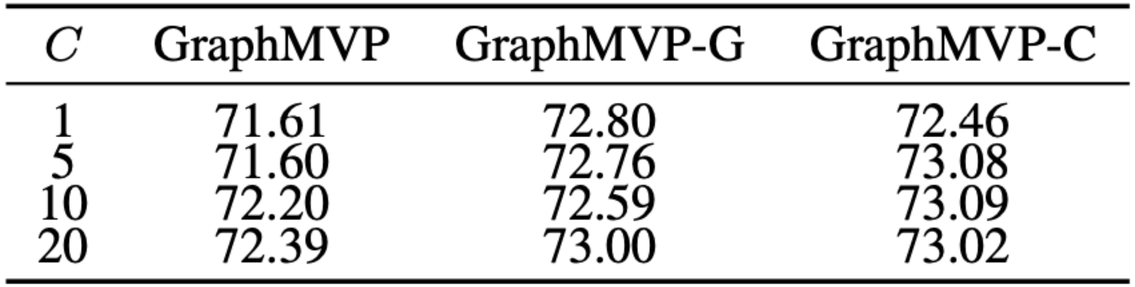
	Molecular Property Prediction					Drug-Target Affinity		
Pre-training	ESOL	Lipo	Malaria	CEP	Avg	Davis	KIBA	Avg
–	1.178	0.744	1.127	1.254	1.0756	0.286	0.206	0.2459
AM	1.112	0.730	1.119	1.256	1.0542	0.291	0.203	0.2476
CP	1.196	0.702	1.101	1.243	1.0606	0.279	0.198	0.2382
JOAO	1.120	0.708	1.145	1.293	1.0663	0.281	0.196	0.2387
GraphMVP	1.091	0.718	1.114	1.236	1.0397	0.280	0.178	0.2286
GraphMVP-G	1.064	0.691	1.106	**1.228**	1.0221	**0.274**	0.175	0.2248
GraphMVP-C	**1.029**	**0.681**	**1.097**	1.244	**1.0128**	0.276	**0.168**	**0.2223**

6. Ablations –

- Effect of #C and Effect of Masking ratio

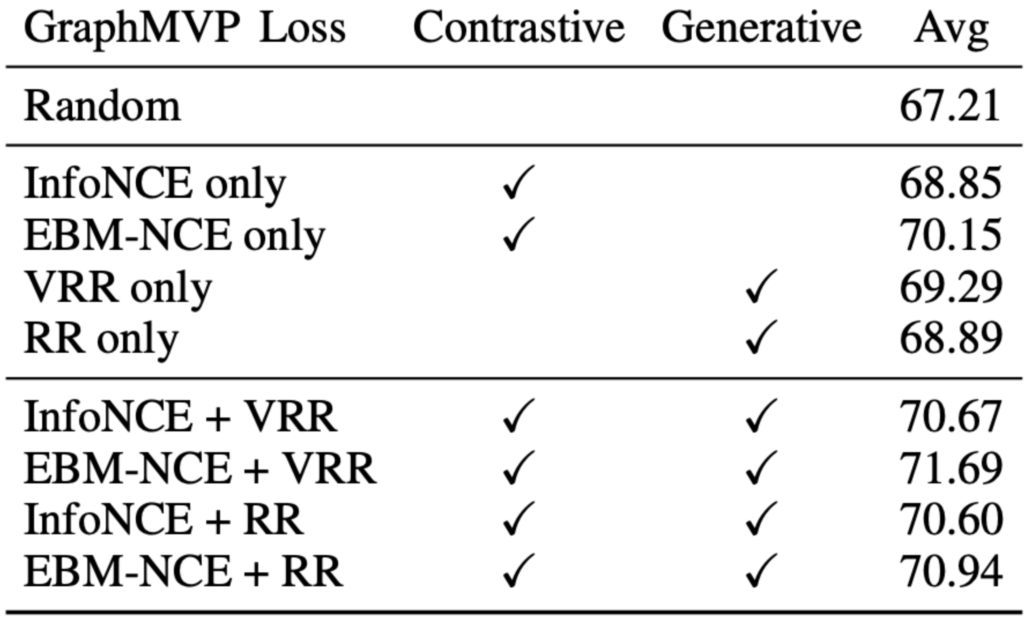


M	GraphMVP	GraphMVP-G	GraphMVP-C
0	71.12	72.15	72.66
0.15	71.60	72.76	73.08
0.30	71.79	72.91	73.17



C	GraphMVP	GraphMVP-G	GraphMVP-C
1	71.61	72.80	72.46
5	71.60	72.76	73.08
10	72.20	72.59	73.09
20	72.39	73.00	73.02

- Effect of Objective function

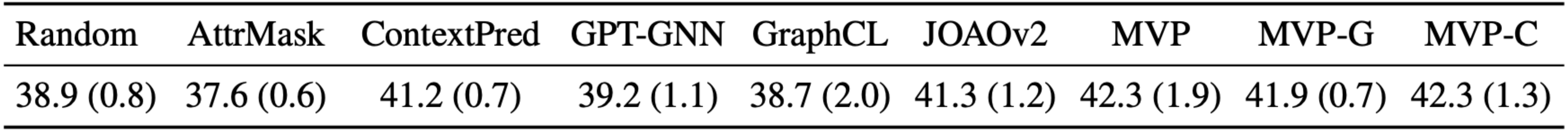


GraphMVP Loss	Contrastive	Generative	Avg
Random			67.21
InfoNCE only	✓		68.85
EBM-NCE only	✓		70.15
VRR only		✓	69.29
RR only		✓	68.89
InfoNCE + VRR	✓	✓	70.67
EBM-NCE + VRR	✓	✓	71.69
InfoNCE + RR	✓	✓	70.60
EBM-NCE + RR	✓	✓	70.94

7. Case study –

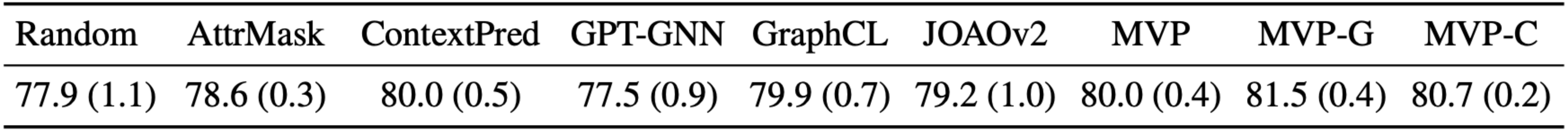
- Evaluate on tasks that are **very difficult** with **only** 2D topology but easy with **3D geometry**.

1. Predicting 3D diameter

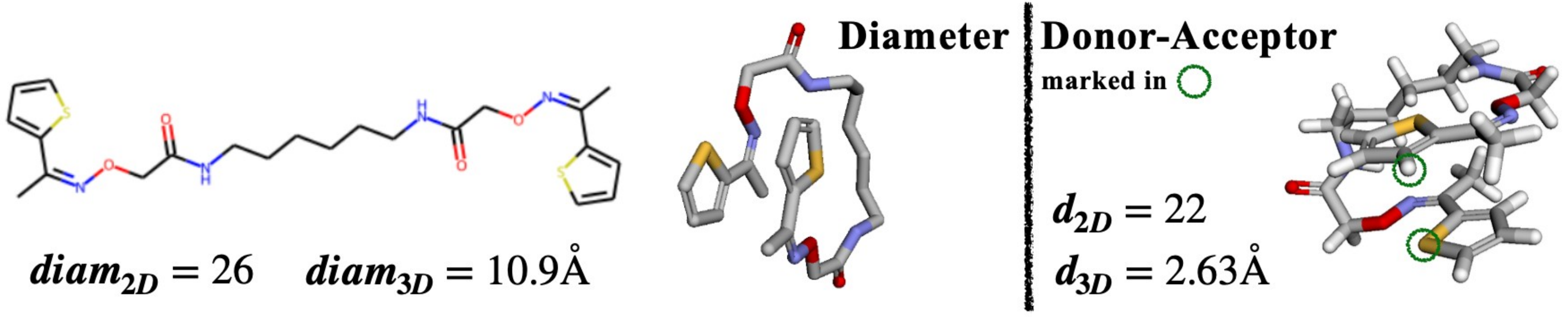


Random	AttrMask	ContextPred	GPT-GNN	GraphCL	JOAOv2	MVP	MVP-G	MVP-C
38.9 (0.8)	37.6 (0.6)	41.2 (0.7)	39.2 (1.1)	38.7 (2.0)	41.3 (1.2)	42.3 (1.9)	41.9 (0.7)	42.3 (1.3)

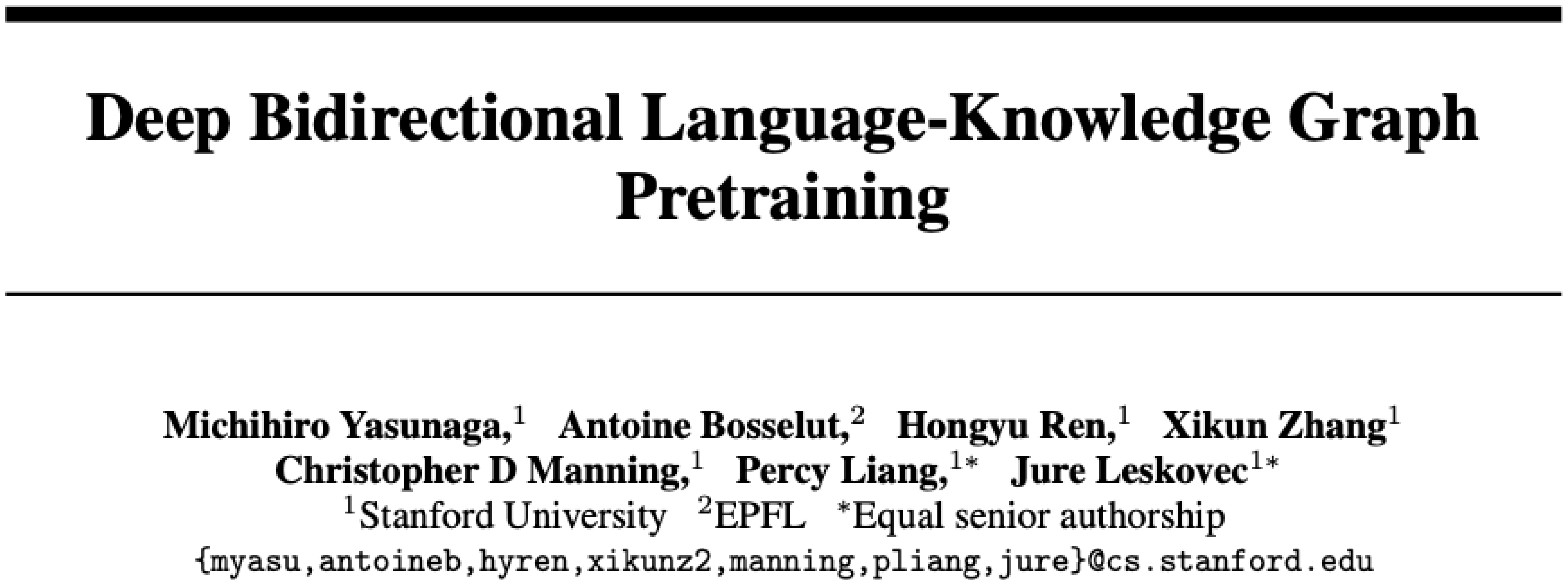
2. Recognizing long-range donor-acceptor structures



Random	AttrMask	ContextPred	GPT-GNN	GraphCL	JOAOv2	MVP	MVP-G	MVP-C
77.9 (1.1)	78.6 (0.3)	80.0 (0.5)	77.5 (0.9)	79.9 (0.7)	79.2 (1.0)	80.0 (0.4)	81.5 (0.4)	80.7 (0.2)



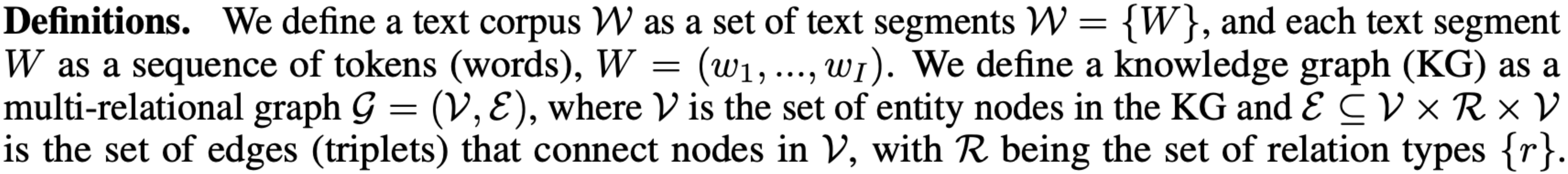
$$R(f) \leq R_n(f) + \mathcal{O}\left(\left(\frac{\mathrm{VCD}(\mathcal{F}) - \log \delta}{n} \right)^{\beta} \right)$$



Deep Bidirectional Language-Knowledge Graph Pretraining

Michihiro Yasunaga,[1] **Antoine Bosselut,**[2] **Hongyu Ren,**[1] **Xikun Zhang**[1]
Christopher D Manning,[1] **Percy Liang,**[1*] **Jure Leskovec**[1*]
[1]Stanford University [2]EPFL [*]Equal senior authorship
{myasu,antoineb,hyren,xikunz2,manning,pliang,jure}@cs.stanford.edu

1. *Introduction* –

- LLMs and large KGs are very useful.
- Open problem to combine both.
- Current work –
  1. Either uses *shallow unidirectional methods,* OR
  2. Focuses on fine-tuning.
- Basic pipeline : Get Text → Extract relevant part of KG → Use a cross-modal model for bidirectional information flow → Use MLM + link prediction as objectives.
- Beats SOTA on various tasks.
- Other related work –
  1. *Knowledge augmented LM pre-training.*
  2. *KG-augmented QA.*
  3. *KG representation learning.*
- 

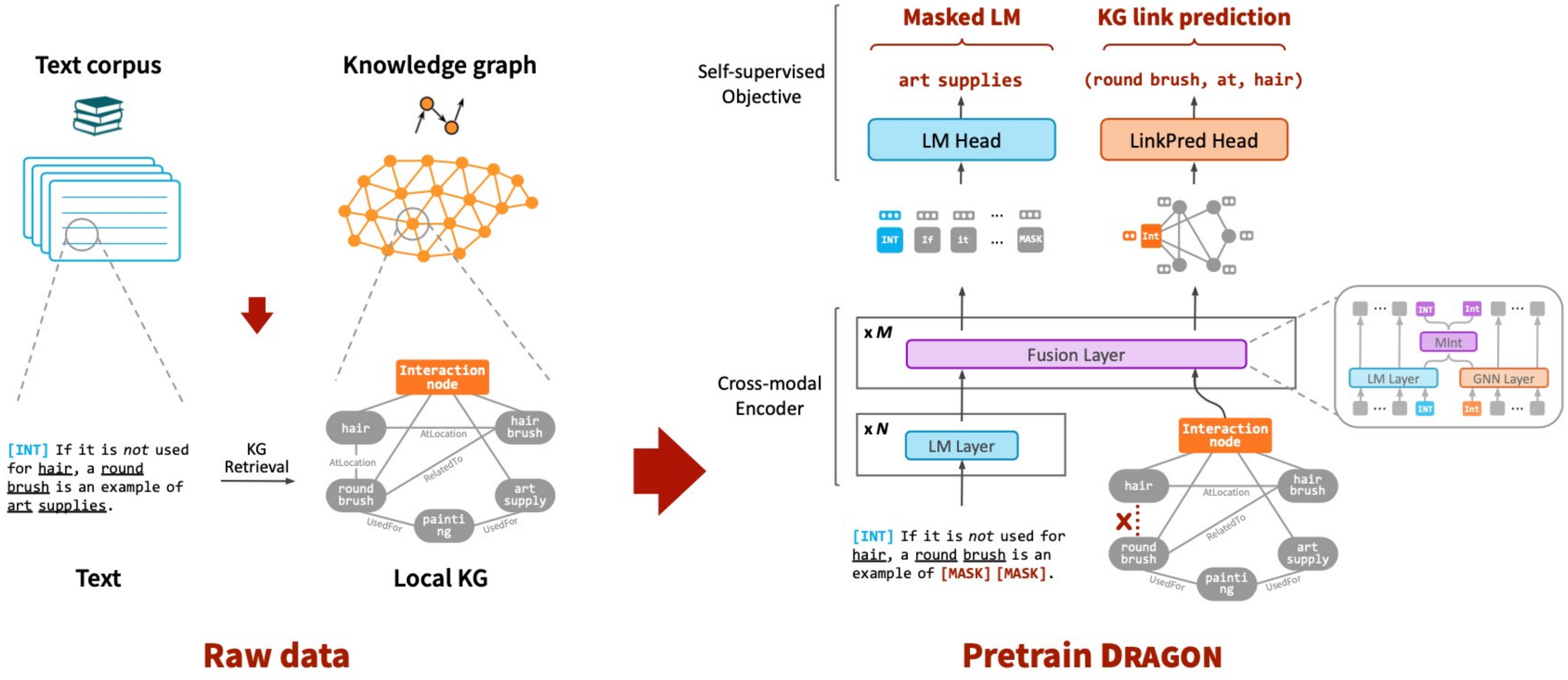
  Definitions. We define a text corpus $\mathcal{W}$ as a set of text segments $\mathcal{W} = \{W\}$, and each text segment W as a sequence of tokens (words), $W = (w_1, ..., w_I)$. We define a knowledge graph (KG) as a multi-relational graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of entity nodes in the KG and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$ is the set of edges (triplets) that connect nodes in $\mathcal{V}$, with $\mathcal{R}$ being the set of relation types $\{r\}$.

2. *Method* –

- First, we need to create the **input representation —**

  1. *Local KG retrieval (V_{el} → V → G),*
  2. *Modality interaction token/node (w_{int} , v_{int} , r_{el}).*

- **Cross-modality encoder** (GreaseLM) –



2. Method –

- **Pre-training objective(s) —**

$$\mathcal{L}_{\text{MLM}} = -\sum_{i \in M} \log p(w_i \mid \mathbf{H}_i)$$

$$\mathcal{L}_{\text{LinkPred}} = \sum_{(h,r,t) \in S} \left(-\log \sigma(\phi_r(\mathbf{h}, \mathbf{t}) + \gamma) + \frac{1}{n} \sum_{(h',r,t')} \log \sigma(\phi_r(\mathbf{h}', \mathbf{t}') + \gamma) \right)$$

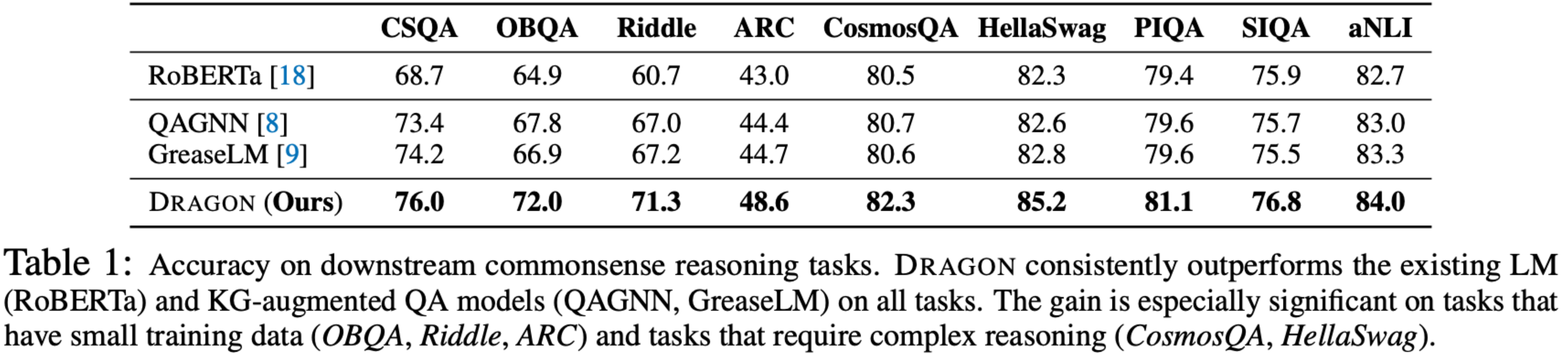
- **Fine-tuning –**

X = MLP(H_{int}, V_{int}, G) ;

G is the attention based pooling of the local KG with **H_{int} as query**.

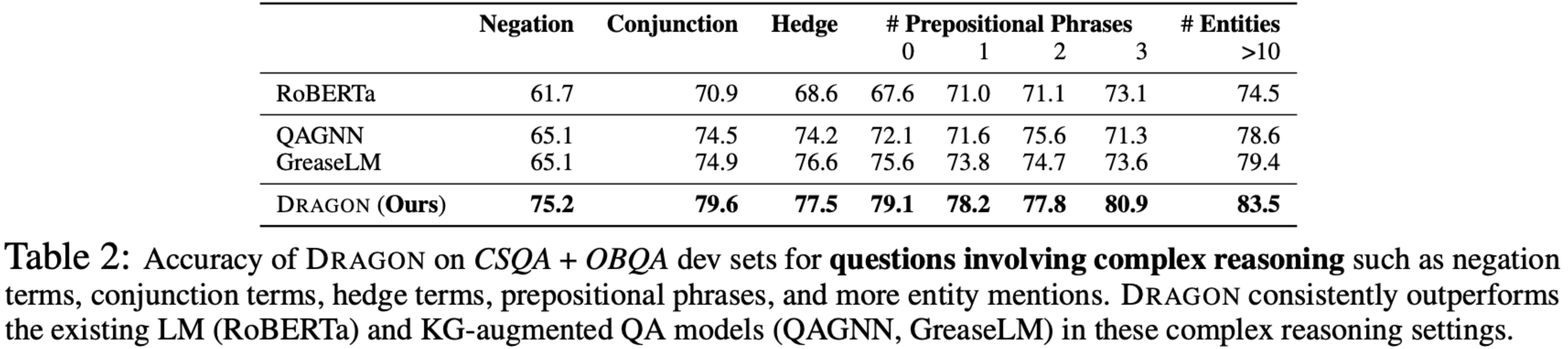
X is used for all downstream tasks.

3. Experiments –



	CSQA	OBQA	Riddle	ARC	CosmosQA	HellaSwag	PIQA	SIQA	aNLI
RoBERTa [18]	68.7	64.9	60.7	43.0	80.5	82.3	79.4	75.9	82.7
QAGNN [8]	73.4	67.8	67.0	44.4	80.7	82.6	79.6	75.7	83.0
GreaseLM [9]	74.2	66.9	67.2	44.7	80.6	82.8	79.6	75.5	83.3
DRAGON (**Ours**)	**76.0**	**72.0**	**71.3**	**48.6**	**82.3**	**85.2**	**81.1**	**76.8**	**84.0**

Table 1: Accuracy on downstream commonsense reasoning tasks. DRAGON consistently outperforms the existing LM (RoBERTa) and KG-augmented QA models (QAGNN, GreaseLM) on all tasks. The gain is especially significant on tasks that have small training data (*OBQA*, *Riddle*, *ARC*) and tasks that require complex reasoning (*CosmosQA*, *HellaSwag*).

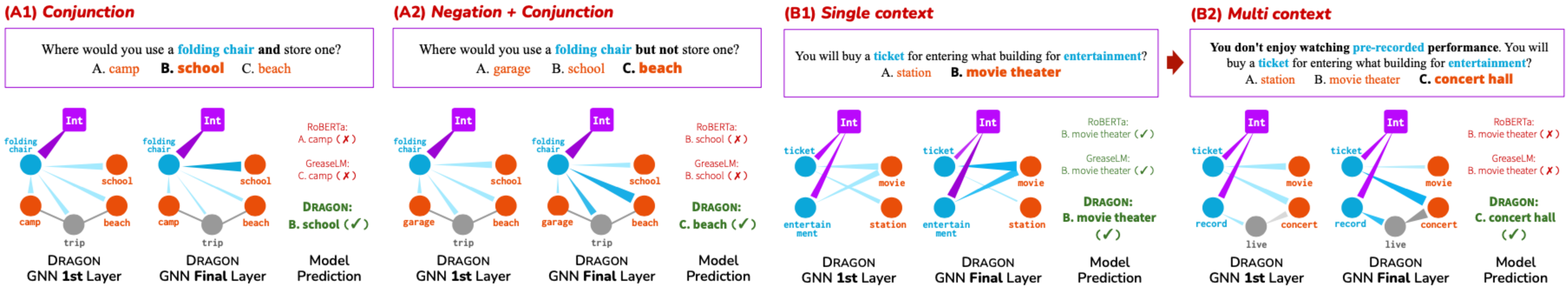


	Negation	Conjunction	Hedge	# Prepositional Phrases				# Entities
				0	1	2	3	>10
RoBERTa	61.7	70.9	68.6	67.6	71.0	71.1	73.1	74.5
QAGNN	65.1	74.5	74.2	72.1	71.6	75.6	71.3	78.6
GreaseLM	65.1	74.9	76.6	75.6	73.8	74.7	73.6	79.4
DRAGON (**Ours**)	**75.2**	**79.6**	**77.5**	**79.1**	**78.2**	**77.8**	**80.9**	**83.5**

Table 2: Accuracy of DRAGON on *CSQA* + *OBQA* dev sets for **questions involving complex reasoning** such as negation terms, conjunction terms, hedge terms, prepositional phrases, and more entity mentions. DRAGON consistently outperforms the existing LM (RoBERTa) and KG-augmented QA models (QAGNN, GreaseLM) in these complex reasoning settings.

4. Analysis –

1. Effect of KG vs LM pre-training.



(A1) *Conjunction*

Where would you use a **folding chair and** store one?
A. camp **B. school** C. beach

RoBERTa: A. camp (✗)
GreaseLM: C. camp (✗)
DRAGON: B. school (✓)

(A2) *Negation + Conjunction*

Where would you use a **folding chair but not** store one?
A. garage B. school **C. beach**

RoBERTa: B. school (✗)
GreaseLM: B. school (✗)
DRAGON: C. beach (✓)

(B1) *Single context*

You will buy a ticket for entering what building for entertainment?
A. station **B. movie theater**

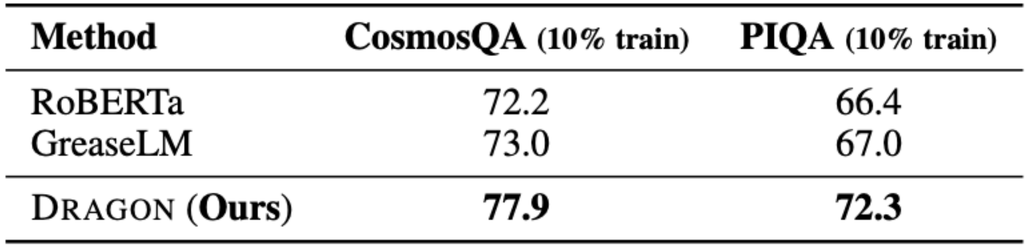
RoBERTa: B. movie theater (✓)
GreaseLM: B. movie theater (✓)
DRAGON: B. movie theater (✓)

(B2) *Multi context*

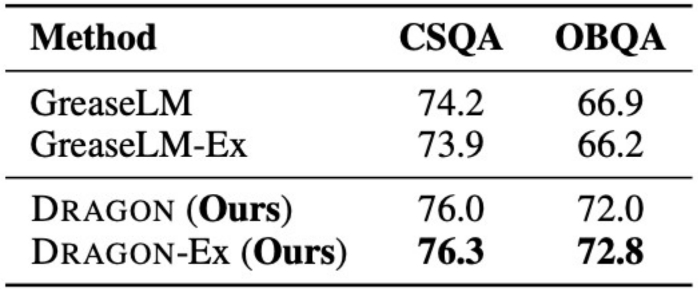
You don't enjoy watching pre-recorded performance. You will buy a ticket for entering what building for entertainment?
A. station B. movie theater **C. concert hall**

RoBERTa: B. movie theater (✗)
GreaseLM: B. movie theater (✗)
DRAGON: C. concert hall (✓)

2. Effect of pre-training vs GreaseLM.



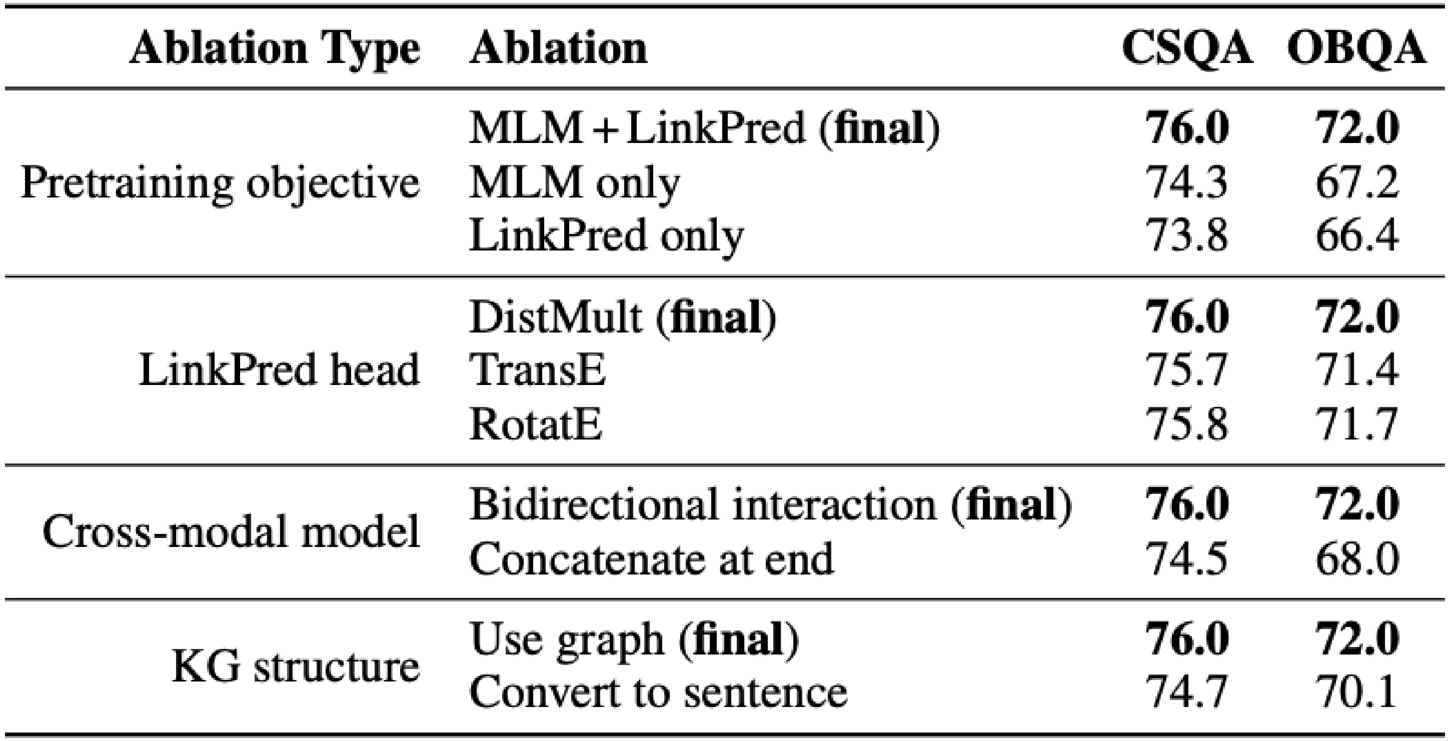
Method	CosmosQA (10% train)	PIQA (10% train)
RoBERTa	72.2	66.4
GreaseLM	73.0	67.0
DRAGON (**Ours**)	**77.9**	**72.3**



Method	CSQA	OBQA
GreaseLM	74.2	66.9
GreaseLM-Ex	73.9	66.2
DRAGON (**Ours**)	76.0	72.0
DRAGON-Ex (**Ours**)	**76.3**	**72.8**

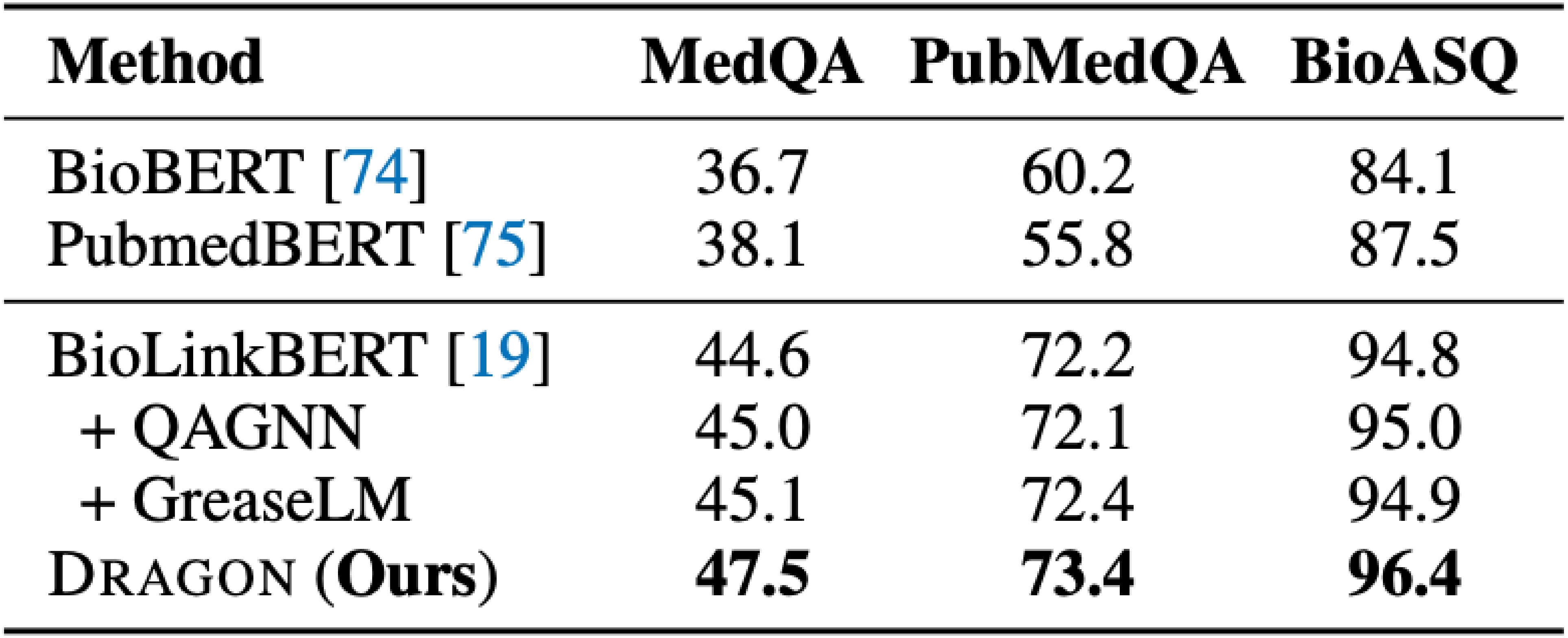
4. <u>Analysis</u> –

3. Ablation



Ablation Type	Ablation	CSQA	OBQA
Pretraining objective	MLM + LinkPred (**final**)	**76.0**	**72.0**
	MLM only	74.3	67.2
	LinkPred only	73.8	66.4
LinkPred head	DistMult (**final**)	**76.0**	**72.0**
	TransE	75.7	71.4
	RotatE	75.8	71.7
Cross-modal model	Bidirectional interaction (**final**)	**76.0**	**72.0**
	Concatenate at end	74.5	68.0
KG structure	Use graph (**final**)	**76.0**	**72.0**
	Convert to sentence	74.7	70.1

4. Bio-medical Experiments –



Method	**MedQA**	**PubMedQA**	**BioASQ**
BioBERT [74]	36.7	60.2	84.1
PubmedBERT [75]	38.1	55.8	87.5
BioLinkBERT [19]	44.6	72.2	94.8
+ QAGNN	45.0	72.1	95.0
+ GreaseLM	45.1	72.4	94.9
Dragon (**Ours**)	**47.5**	**73.4**	**96.4**