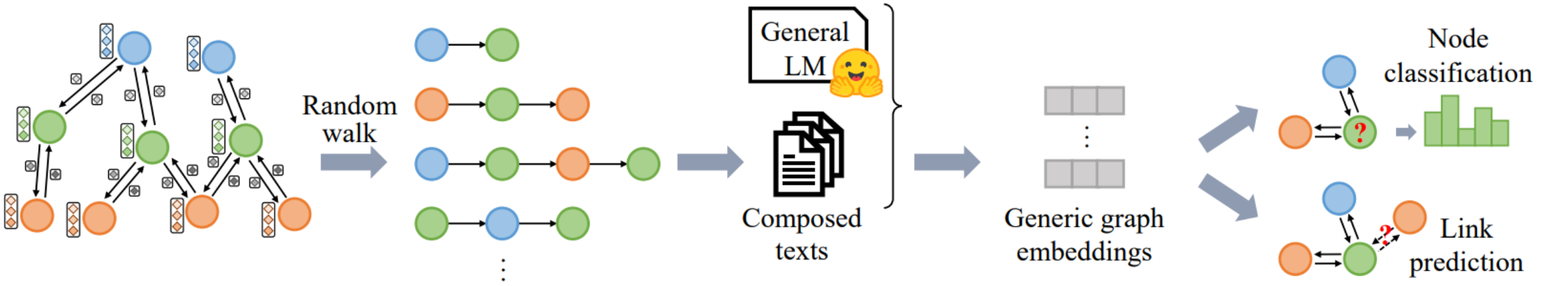
All-hands Meeting III
2024 Winter

WalkLM: A Uniform Language Model Fine-tuning Framework for Attributed Graph Embedding

Presenter: Minseo Yoon
(cooki0615@korea.ac.kr)

Recap



(a) Attributed RW-based textualization program

(b) Graph-aware LM fine-tuning

(c) Downstream tasks

$$\Phi(v_i) = \{age : 35,\ sex : \text{female},\ pid : \text{P246}\}$$

$$\mathcal{P}(v_i) = < \text{A 35-year-old female patient P246} >$$

$$W = \{\mathcal{P}(v_0), \mathcal{P}(e_1), \mathcal{P}(v_1), \ldots, \mathcal{P}(v_{L-1}), \mathcal{P}(e_L), \mathcal{P}(v_L)\}$$

$$\mathcal{T} = \{t_1, t_2, \ldots, t_K, < \text{MASK} >_1, < \text{MASK} >_2, \ldots, < \text{MASK} >_{|\mathcal{M}|}\}$$

$$\mathcal{L}_{FT}(\Theta) = -\frac{1}{|\mathbb{X}|} \sum_{X_i \in \mathbb{X}} \left[\sum_{t_k^* \in \mathcal{M}} \log \frac{\exp(Sim(t_k, t_k^*))}{\sum_{t \in \mathcal{V}} \exp(Sim(t_k, t))} \right]$$

Methodology

Attributed RW-based textualization program

- Textualization
  - Rule-based program function $\mathcal{P}(\cdot)$
  - Random walk $W = \{\mathcal{P}(v_0), \mathcal{P}(e_1), \mathcal{P}(v_1), \ldots, \mathcal{P}(v_{L-1}), \mathcal{P}(e_L), \mathcal{P}(v_L)\}$

  - 기존 Edge: in, and, with, causing 중 하나
  - 개선된 Edge: GPT-4 generated rule-based edge
    - is associated with, leads to, causes, is diagnosed by, is prevalent in, ⋯
    - 단어의 연결성과 문맥을 고려해서 edge를 생성해달라고 prompting

Methodology

Graph-aware LM fine-tuning

- 학습 방식 개선
  1. 원래보다 3배 짧고 45배 적은 random walk만을 사용해서 학습
  2. Graph의 structure knowledge를 사용하기 위해 language model의 input으로 random masking이 아닌 edge masking을 사용

```
Source: [Node 1, _____, Node 2, _____, Node 3, _____, ..., ________, Node N]
Target: [Node 1, Edge 1, Node 2, Edge 2, Node 3, Edge 3, ..., Edge N-1, Node N]
```

  3. 기존에 사용한 Encoder based LM: DistilRoBERTa
     → Seq2Seq 방식인 Encoder-Decoder based LM T5-XL을 fine-tuning

     Trainable parameters: 83M → 2M

```
model_name = "google/flan-t5-xl"

config = LoraConfig(
    r=4,
    lora_alpha=8,
    target_modules=['q', 'v'],
    bias="none",
    lora_dropout=0.1,
    task_type="SEQ_2_SEQ_LM",
)
```

Methodology

Graph-aware LM fine-tuning

- 학습 방식 개선
  4. 기존에 사용한 Encoder based LM: DistilRoBERTa
     → (head, ?, tail)뿐만 아니라 (head, predicate, ?)을 학습하는 것도 그래프 구조를 이용하는 것
     → Decoder based LM Llama2 7B를 fine-tuning

     Trainable parameters: 83M → 2M
  - 특이사항: Colab(A100 40GB)에서 구동하기 위하여 4-bit quantization을 거친 model을 load

```
bnb_config = BitsAndBytesConfig(
    load_in_4bit=True,
    bnb_4bit_use_double_quant=True,
    bnb_4bit_quant_type="nf4",
    bnb_4bit_compute_dtype=torch.bfloat16
)
```

Methodology

Downstream tasks

- Preliminary: JK-Nets
  - Random walk distribution

Definition 3.2. *Consider a random walk on $\widetilde{G}$ starting at a node v_0; if at the t-th step we are at a node v_t, we move to any neighbor of v_t (including v_t) with equal probability. The t-step* random walk distribution *P_t of v_0 is*

$$P_t(i) = Prob(v_t = i).$$

Analogous definitions apply for random walks with non-uniform transition probabilities.

Representation Learning on Graphs with Jumping Knowledge Networks

$$p_v^{t+1} = \sum_{e \in E, e=(u,v)} \frac{1}{d(u)} p_u^t \quad \text{for each } v \in V$$

https://people.csail.mit.edu/madry/gems/notes/lecture9.pdf

Methodology
Downstream tasks

- Preliminary: JK-Nets
  - Random walk distribution
  - Idea: Neighborhood Aggregation with Skip Connections
  1. Typical neighborhood aggregation scheme

$$h_v^{(l)} = \sigma\left(W_l \cdot \text{AGGREGATE}\left(\{h_u^{(l-1)}, \forall u \in \tilde{N}(v)\}\right)\right) \qquad \tilde{N}(v) = \{v\} \cup \{u \in V | (v,u) \in E\}$$

  2. Neighborhood aggregation with Skip Connections

$$h_{N(v)}^{(l)} = \sigma\left(W_l \cdot \text{AGGREGATE}_N\left(\{h_u^{(l-1)}, \forall u \in N(v)\}\right)\right)$$

$$h_v^{(l)} = \text{COMBINE}\left(h_v^{(l-1)}, h_{N(v)}^{(l)}\right) \qquad N(v) = \{u \in V | (v,u) \in E\}$$

  - 그러나 단순히 skip하기만 하는 것은 최종 reasoning 단계에 대해 adaptive한 방법이 될 수 없음

Representation Learning on Graphs with Jumping Knowledge Networks

Methodology
Downstream tasks

- Preliminary: JK-Nets
  - Random walk distribution
  - Idea: Neighborhood Aggregation with Skip Connections

$$h_{N(v)}^{(l)} = \sigma\left(W_l \cdot \text{AGGREGATE}_N\left(\{h_u^{(l-1)}, \forall u \in N(v)\}\right)\right)$$
$$h_v^{(l)} = \text{COMBINE}\left(h_v^{(l-1)}, h_{N(v)}^{(l)}\right)$$

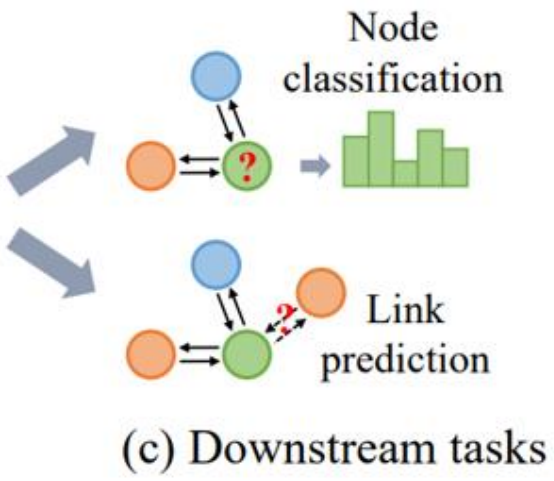
  - 만약 skip을 확률적으로 한다면? 또는 확률을 학습한다면?
    - Graph Convolutional Networks vs. GCN with residual connections
    - Random walk vs. RW with lazy factor
    - 본 연구에서는 이웃 node의 정보를 aggregate하는 GCN의 학습 방식과 이웃한 node 사이의 edge를 예측하거나 이웃 node를 예측하는 방식으로 fine-tuning된 LM의 embedding 사이의 유사성을 찾아볼 수 있음

Representation Learning on Graphs with Jumping Knowledge Networks

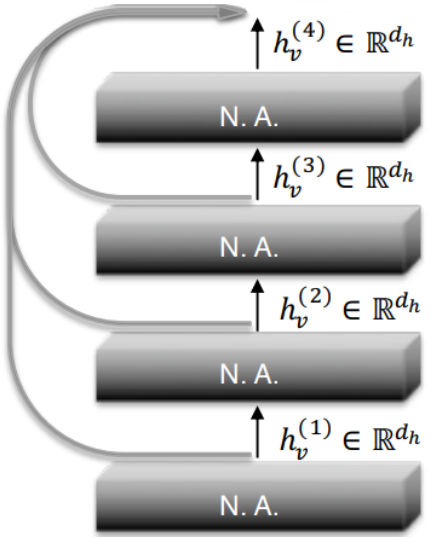
Methodology

Downstream tasks

- Analogy between multi-step random walk and skip connection
  - 기존: 단순히 MLP를 통해 최종 classification을 진행
  - 개선: Skip connection을 도입하여 adaptivity를 증진



+



Experiments

Language Model				Node Classification		Link Prediction	
Name	GPT-4 Rules	Edge Masking	Skip Conn.	Macro-F1	Micro-F1	AUC	MRR
DistilRoBERTa (Baseline)				60.42	62.33	85.65	94.16
Flan-T5-XL	O			74.30	74.68	87.09	94.76
Flan-T5-XL		O		73.59	74.45	**87.58**	**95.21**
Flan-T5-XL	O	O		74.09	74.67	87.30	95.12
Flan-T5-XL		O	O	75.39	76.21	85.84	94.66
Flan-T5-XL	O	O	O	**75.77**	**76.65**	85.86	94.91
Llama2-7B-Chat		No input		**77.13**	**77.97**	**82.24**	**94.38**
Llama2-7B-Chat	O	No input		75.01	75.76	80.92	93.42
Llama2-7B-Chat		No input	O	75.66	77.30	81.05	93.87
Llama2-7B-Chat	O	No input	O	76.91	**77.97**	80.77	93.81

- Encoder-Decoder based LM (T5)
  1. Node classification에서 GPT-4가 생성한 rule에 의해 성능이 평균적으로 0.4% 상승
  2. Link Prediction에서는 미미한 차이
  - 복잡한 relation을 추론하도록 하는 것이 소규모 LLM에게 오히려 방해 요소로 작용할 수 있음

Experiments

Language Model				Node Classification		Link Prediction	
Name	GPT-4 Rules	Edge Masking	Skip Conn.	Macro-F1	Micro-F1	AUC	MRR
DistilRoBERTa (Baseline)				60.42	62.33	85.65	94.16
Flan-T5-XL	O			74.30	74.68	87.09	94.76
Flan-T5-XL		O		73.59	74.45	**87.58**	**95.21**
Flan-T5-XL	O	O		74.09	74.67	87.30	95.12
Flan-T5-XL		O	O	75.39	76.21	85.84	94.66
Flan-T5-XL	O	O	O	**75.77**	**76.65**	85.86	94.91
Llama2-7B-Chat		No input		**77.13**	**77.97**	**82.24**	**94.38**
Llama2-7B-Chat	O	No input		75.01	75.76	80.92	93.42
Llama2-7B-Chat		No input	O	75.66	77.30	81.05	93.87
Llama2-7B-Chat	O	No input	O	76.91	**77.97**	80.77	93.81

- Encoder-Decoder based LM (T5)
  1. Node classification에서 Edge masking에 의해 성능이 평균적으로 0.5% 하락
  2. Link Prediction에서는 평균적으로 0.5% 상승
  - Edge에 해당하는 sequence를 추론하도록 fine-tuning한 LM의 embedding이 추론에 도움이 됨

Experiments

Language Model				Node Classification		Link Prediction	
Name	GPT-4 Rules	Edge Masking	Skip Conn.	Macro-F1	Micro-F1	AUC	MRR
DistilRoBERTa (Baseline)				60.42	62.33	85.65	94.16
Flan-T5-XL	O			74.30	74.68	87.09	94.76
Flan-T5-XL		O		73.59	74.45	**87.58**	**95.21**
Flan-T5-XL	O	O		74.09	74.67	87.30	95.12
Flan-T5-XL		O	O	75.39	76.21	85.84	94.66
Flan-T5-XL	O	O	O	**75.77**	**76.65**	85.86	94.91
Llama2-7B-Chat		No input		**77.13**	**77.97**	**82.24**	**94.38**
Llama2-7B-Chat	O	No input		75.01	75.76	80.92	93.42
Llama2-7B-Chat		No input	O	75.66	77.30	81.05	93.87
Llama2-7B-Chat	O	No input	O	76.91	**77.97**	80.77	93.81

- Encoder-Decoder based LM (T5)
  1. Node classification에서 Skip connection에 의해 성능이 평균적으로 1.8% 상승
  2. Link Prediction에서는 성능 유지
  - 다양한 정보를 고려하는 것이 성능 향상에 도움이 될 수 있음

Experiments

Language Model				Node Classification		Link Prediction	
Name	GPT-4 Rules	Edge Masking	Skip Conn.	Macro-F1	Micro-F1	AUC	MRR
DistilRoBERTa (Baseline)				60.42	62.33	85.65	94.16
Flan-T5-XL	O			74.30	74.68	87.09	94.76
Flan-T5-XL		O		73.59	74.45	**87.58**	**95.21**
Flan-T5-XL	O	O		74.09	74.67	87.30	95.12
Flan-T5-XL		O	O	75.39	76.21	85.84	94.66
Flan-T5-XL	O	O	O	**75.77**	**76.65**	85.86	94.91
Llama2-7B-Chat		No input		**77.13**	**77.97**	**82.24**	**94.38**
Llama2-7B-Chat	O	No input		75.01	75.76	80.92	93.42
Llama2-7B-Chat		No input	O	75.66	77.30	81.05	93.87
Llama2-7B-Chat	O	No input	O	76.91	**77.97**	80.77	93.81

- Decoder based LM (Llama2)
  1. 다양한 방법론을 적용할수록 성능이 하락하는 경향을 보임
  - Decoder의 embedding이 generation에 특화되어 있기 때문에 추가적인 방법이 개입되는 것이 방해 요소로 작용할 수 있음

Experiments

Language Model				Node Classification		Link Prediction	
Name	GPT-4 Rules	Edge Masking	Skip Conn.	Macro-F1	Micro-F1	AUC	MRR
DistilRoBERTa (Baseline)				60.42	62.33	85.65	94.16
Flan-T5-XL	O			74.30	74.68	87.09	94.76
Flan-T5-XL		O		73.59	74.45	**87.58**	**95.21**
Flan-T5-XL	O	O		74.09	74.67	87.30	95.12
Flan-T5-XL		O	O	75.39	76.21	85.84	94.66
Flan-T5-XL	O	O	O	**75.77**	**76.65**	85.86	94.91
Llama2-7B-Chat		No input		**77.13**	**77.97**	**82.24**	**94.38**
Llama2-7B-Chat	O	No input		75.01	75.76	80.92	93.42
Llama2-7B-Chat		No input	O	75.66	77.30	81.05	93.87
Llama2-7B-Chat	O	No input	O	76.91	**77.97**	80.77	93.81

- 하나를 선택해야 한다면?
  - Node classification과 Link prediction에서 균형 있게 좋은 성능을 보이고 pre-trained parameter가 적으며 inference time이 상대적으로 짧은 T5를 선택할 것

Qualitative Results

```
Masked text: LRRC26 <extra_id_0> <extra_id_0> <extra_id_0> <extra_id_0> SIV-infected
True text: is a cause of
Predicted text: LRRC26 is a protein that is expressed in the lungs of SIV-
```

- T5 내의 지식으로 오히려 더 적절한 설명을 하는 경우가 존재

```
Masked text: 5-FUR <extra_id_0> <extra_id_0> <extra_id_0> Daidzin <extra_id_0> <extra_id_0> <extra_id_0> Vav1
True text: is observed in is detected in
Predicted text: 5-FUR is a genus of flowering plants in the family Rubiace
```

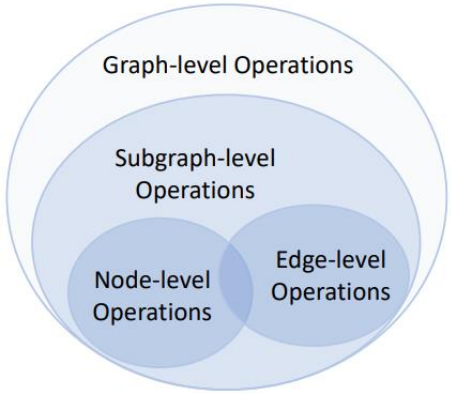
- 물론 소규모 LLM의 특성 상 아예 다른 이야기를 하는 경우도 있음

Limitation / Future Work

- 더욱 다양한 조합을 고려하여 table을 구성하는 것에 대한 시간적 여유 부족
  - Skip connection을 떠올리고 나서부터 발표 자료 제작까지의 촉박한 시간에 기인함
- 실험 환경의 한계
  - Quantization이 성능에 영향을 미칠 수 있음
  - 하나의 GPU로 LLM을 학습해야 했기 때문에 T5는 3 epoch, Llama2는 2 epoch를 적은 양의 데이터로 학습하는 데 그침

Limitation / Future Work

- GPT-4 generated rule의 한계
  - Prompt에 텍스트 파일 전체를 한 번에 넣어주는 방식으로 target text의 edge를 구성하였기 때문에 문맥을 거의 고려하지 못함
    - API를 통해 문장 단위로 입력하여 해결할 수 있음
    - 고전적인 NLP 알고리즘에서 더욱 robust한 방향성을 얻을 수 있을 것으로 예상
      (더욱 큰 corpus에서 구성한 co-occurrence matrix의 statistic, TF-IDF 등을 이용한 rule 구성)
  - PubMed dataset이 의학적 지식으로 구성되어 있기 때문에 GPT-4가 대답을 회피하려는 경향이 큼
- Graph-level task에 대한 실험도 진행해 볼 수 있을 것으로 예상
  - Node classification, Link prediction은 Subgraph-level task의 부분 집합에 속함



Questions?