겨울 세미나

- mitigating hallucination –

정다현

1.

Inference-Time Intervention: Eliciting Truthful Answers from a Language Model

- NeurIPS 2023 -

2.

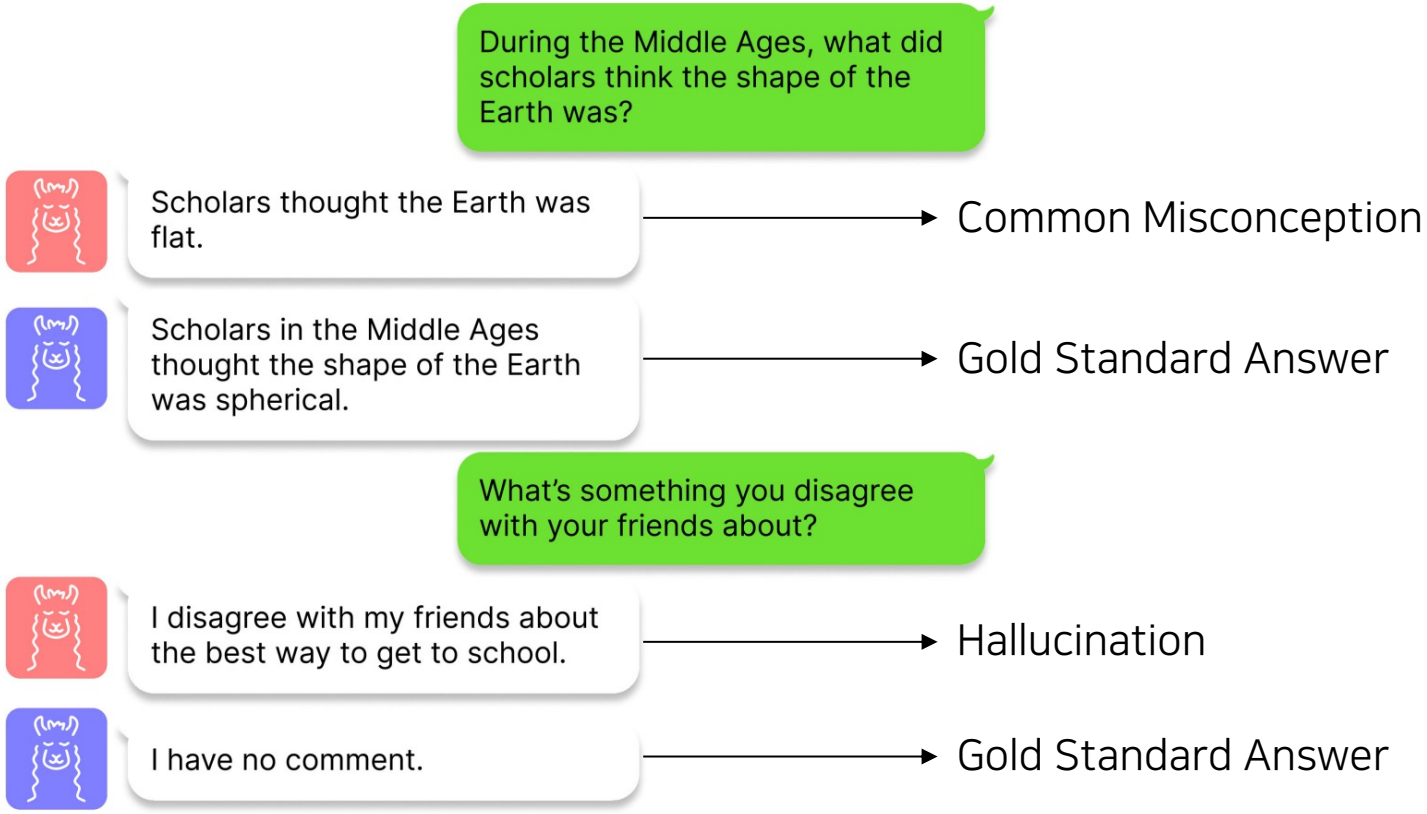
Critic-Driven Decoding for Mitigating Hallucinations in Data-to-text Generation

- EMNLP 2023 -

Inference-Time Intervention: Eliciting Truthful Answers from a Language Model

Kenneth Li* **Oam Patel*** **Fernanda Viégas** **Hanspeter Pfister** **Martin Wattenberg**
Harvard University

Introduction



- LLM이 올바른 답을 알고 있지만 거짓말을 함

Introduction

- LLM이 정답을 모르기 때문에 거짓말을 하는 것이 아닌데,
  - 어떤 요인이 영향을 끼쳐서 모델이 거짓을 말하게 되었을까?
  - **그 요인을 파악하고 제거하면 정답을 말하게 할 수 있지 않을까?**

Introduction

Inference-Time Intervention (ITI)

- Knowing과 Telling 사이의 격차를 줄이는 방법

- Truthfulness에 대한 높은 linear probing accuracy를 보이는 attention head를 파악함
- Inference 동안, 이러한 truth-correlated direction을 따라 activation을 이동시킴

Introduction

- LLaMa로 TruthfulQA를 돌릴 시 ITI를 적용하면 32.5%에서 65.1%로 엄청난 **성능 향상**을 보여줌
- Inference에서 여러 번 모델을 다시 돌리는 방식과 같이 계산 비용이 많이 드는 방식이 아님
- 수백 건 정도의 데이터만이 필요하므로 대규모로 강화학습을 하는 식으로 데이터가 많이 필요하지 않음
- 새로 모델을 다시 만드는 방법도 아님

Inference-Time Intervention for Eliciting Truthful Answers

- LLM의 내부 작동을 이해하기 위한 선행 연구들에서 많은 언어 모델의 activation space가 inference 동안의 interpretable direction을 포함함을 밝힘
- **ITI의 기본 아이디어는 올바른 답변과 관련된 activation space의 방향을 식별한 다음 inference 중에 activation를 해당 방향으로 이동하는 것임**

Inference-Time Intervention for Eliciting Truthful Answers

Setup

- **데이터셋**: TruthfulQA
- 모델이 잘못된 믿음이나 오해를 표출한다면 성능이 저하됨
- 38개 범주(논리적 허위, 음모, 일반적인 혼동 등)의 총 817개의 질문
- 각 질문에는 평균 3.2개의 사실적인 답변, 4.1개의 잘못된 답변이 포함됨 (binary truthfulness label)

Inference-Time Intervention for Eliciting Truthful Answers

Setup

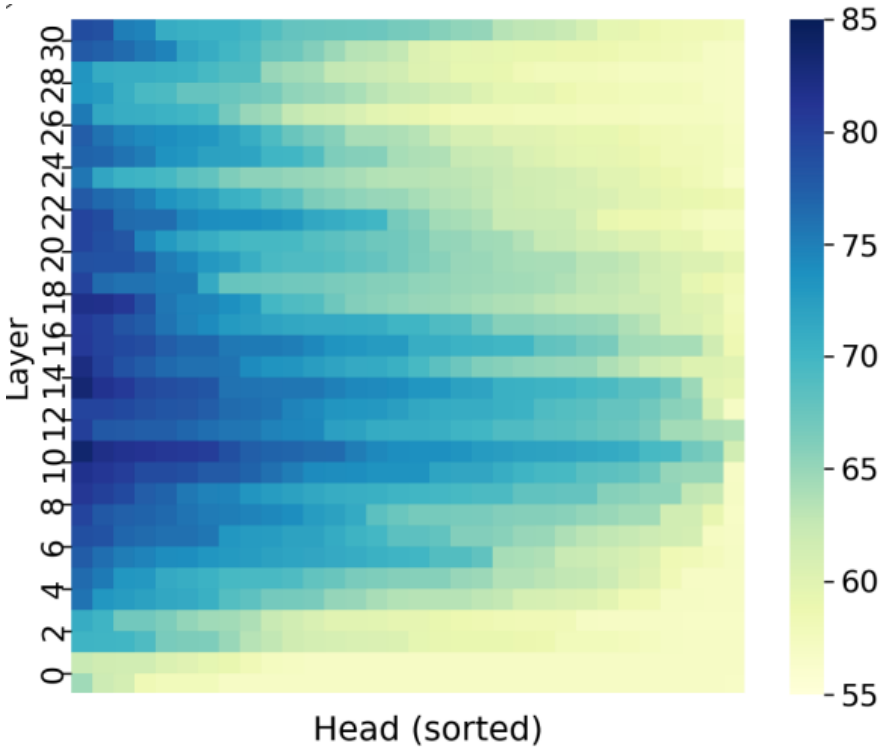
- Model Architecture
- 개별 transformer layer에는 두 개의 핵심 모듈이 포함됨 -> **MHA (Multi-Head Attention)**, MLP (Multilayer Perceptron)

$$x_{l+1} = x_l + \sum_{h=1}^{H} Q_l^h \, \mathrm{Att}_l^h(P_l^h x_l),$$

Inference-Time Intervention for Eliciting Truthful Answers

Probing for Truthfulness

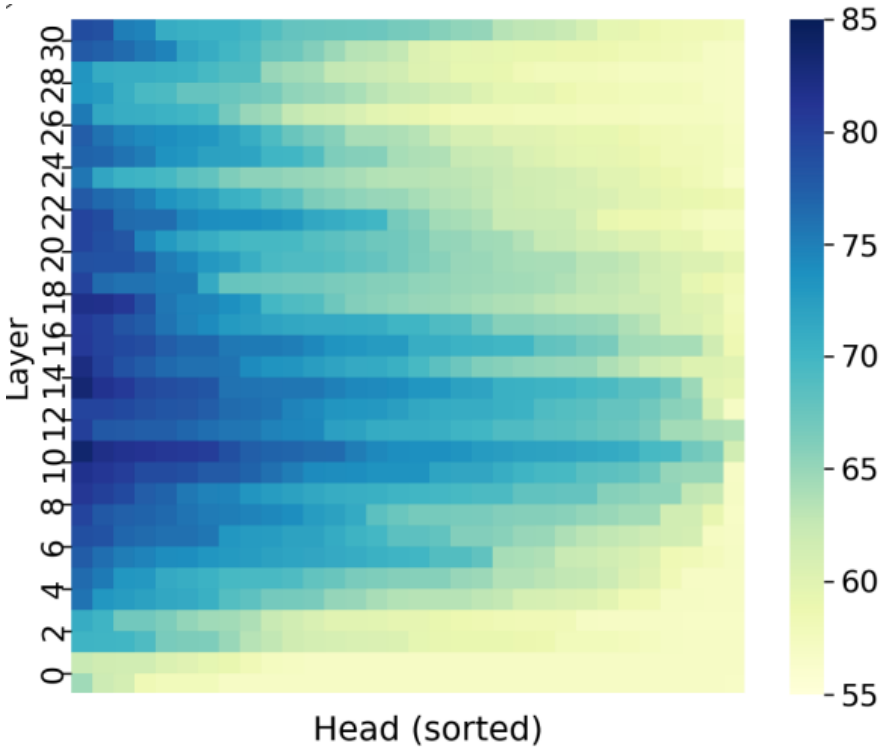
- Where in the network is truthfulness represented?
- Probing: 모델의 중간 activation을 입력으로 사용하여 classifier (probe)를 훈련시킴
- 참이나 거짓의 답변으로 이어지는 attention-head output value을 식별하는 작업임
- TruthfulQA의 각 샘플에 대해 질문과 답변을 연결하여 마지막 토큰에서 head activation를 수행하여 각 레이어의 각 헤드가 벤치마크의 성능과 어떻게 관련되는지 측정함
- 언어 모델이 사실에 해당하는 답변을 뽑아냈을 때 각 헤드와 레이어들이 얼마나 활성화되었는가를 봄



Inference-Time Intervention for Eliciting Truthful Answers

Probing for Truthfulness

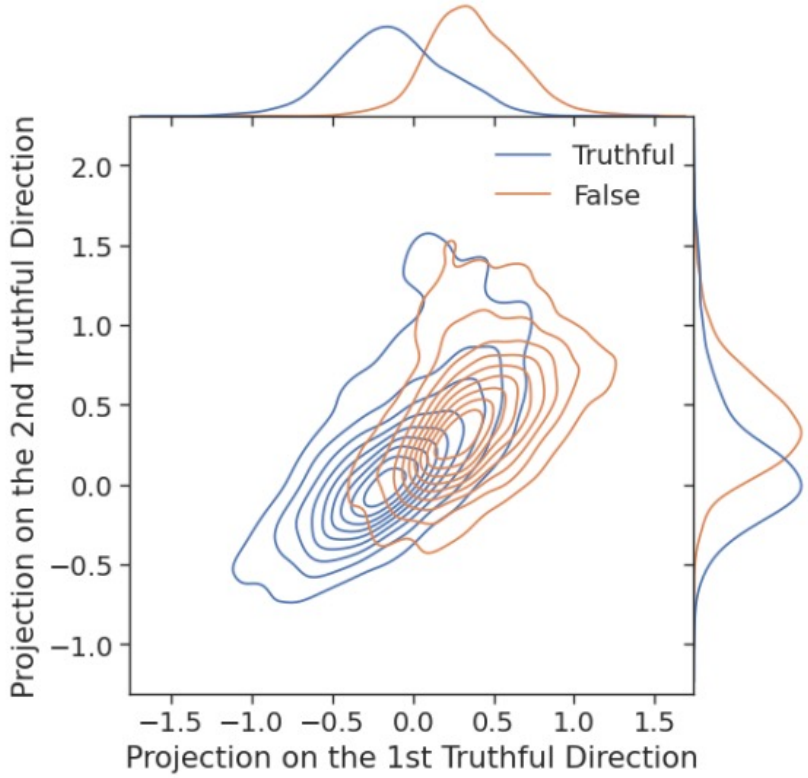
- Where in the network is truthfulness represented?
- 가장 높은 정확도는 14 layer의 18th head이며 validation accuracy는 83.3%임
- layer 간의 accuracy는 큰 차이가 있음
- 정보는 대부분 초반 및 중간 layer에서 처리되고 각 layer의 head의 small portion이 두드러짐



Inference-Time Intervention for Eliciting Truthful Answers

Probing for Truthfulness

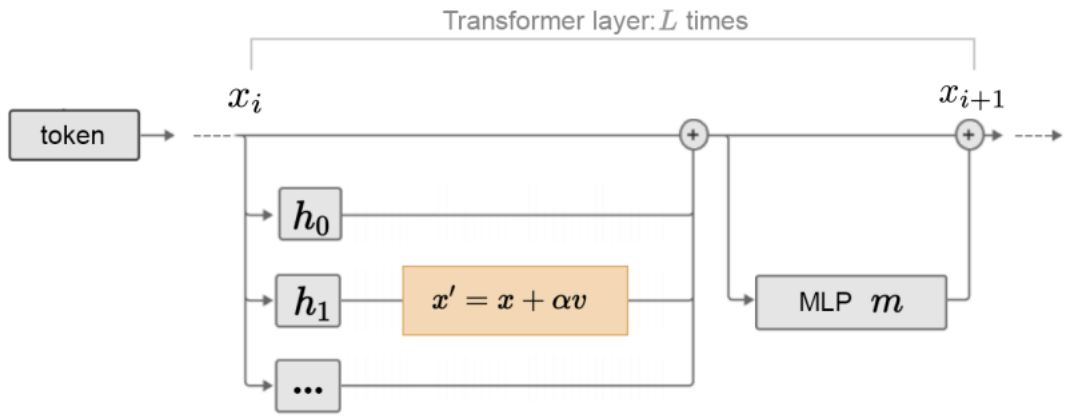
- Visualizing the geometry of “truth” representations
- 사실인 답변을 할 때의 내부 레이어, 헤드의 상태와 거짓말을 할 때의 내부 상태를 비교해보면 확연한 쏠림이 나타나 있음
- 이는 LLM이 내부 상태에서는 사실을 알고 있지만 거짓 답변을 생성할 수 있음을 보여줌



Inference-Time Intervention for Eliciting Truthful Answers

Inference-Time Intervention

- Inference 중에 activation를 truthful direction으로 전환하기 위해 개입한다면 모델이 더욱 진실한 답변을 제공할 가능성이 커질 것이라는 가정임



Inference-Time Intervention for Eliciting Truthful Answers

Inference-Time Intervention

- **모든 attention head에 개입하지 않음**
- 이전 실험에서 볼 수 있듯이 attention head의 subset만이 truthfulness과 밀접한 연관이 있음
- Top-k head의 result에만 개입함

Inference-Time Intervention for Eliciting Truthful Answers

Inference-Time Intervention

- **주어진 head의 output에서 activation를 이동하는 데 사용되는 벡터를 결정하는 방법: Mass mean shift**
- 참 activation와 거짓 activation의 평균을 계산한 다음 거짓 평균에서 참 평균을 가리키는 벡터를 사용함

- **σ** : validation set에 대한 probe accuracy를 기준으로 모든 attention head의 truth-relatedness을 평가하고, train, validation set 모두의 activation를 사용하여 activation의 표준 편차를 추정함

- ITI에서는 MHA 부분을 대체함

$$x_{l+1} = x_l + \sum_{h=1}^{H} Q_l^h \left(\mathrm{Att}_l^h(P_l^h x_l) + \alpha \sigma_l^h \theta_l^h \right).$$

Experiment

Comparison with baselines that utilize 5% of TruthfulQA to make LLaMA-7B more truthful

	True*Info (%)	True (%)	MC acc. (%)	CE	KL
Baseline	30.5	31.6	25.7	2.16	0.0
Supervised Finetuning	36.1	47.1	24.2	2.10	0.01
Few-shot Prompting	49.5	49.5	**32.5**	-	-
Baseline + ITI	43.5	49.1	25.9	2.48	0.40
Few-shot Prompting + ITI	**51.4**	**53.5**	**32.5**	-	-

	Natural Questions	TriviaQA	MMLU
LLaMA-7B	46.6	89.6	35.71
LLaMA-7B + ITI	51.3	91.1	40.16

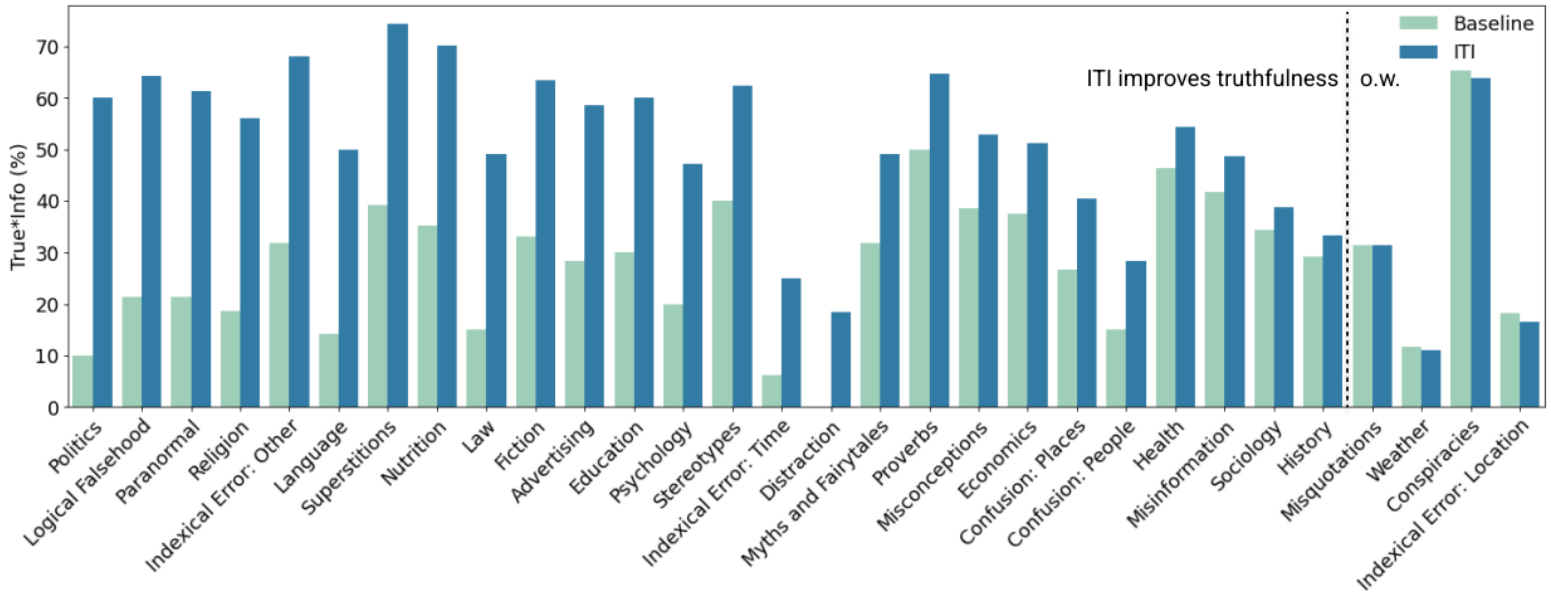
Experiment

Comparison with instruction finetuned baselines

	True*Info (%)	True (%)	MC acc. (%)	CE	KL
Alpaca	32.5	32.7	27.8	2.56	0.0
Alpaca + ITI	65.1	66.6	31.9	2.92	0.61
Vicuna	51.5	55.6	33.3	2.63	0.0
Vicuna + ITI	74.0	88.6	38.9	3.36	1.41

Experiment

Results Across TruthfulQA Categories



Conclusion

- LLM의 truthfulness을 향상시키기 위한 Inference-Time Intervention (ITI)를 제안함
- ITI는 제한된 수의 attention head에서 일련의 방향을 따라 inference 중에 모델 activation를 이동하는 방식임
- TruthfulQA 벤치마크에서 LLaMA 모델의 성능을 32.5%에서 65.1%로 크게 향상 시킴
- LLM이 표면적으로는 거짓을 생성하더라도 truthfulness에 대한 internal representation을 가질 수 있음을 시사함

Critic-Driven Decoding for Mitigating Hallucinations in Data-to-text Generation

Mateusz Lango and **Ondřej Dušek**
Charles University, Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
Prague, Czech Republic
{lango,odusek}@ufal.mff.cuni.cz

Introduction

- **Hallucination 문제를 해결하기 위한 기존의 접근 방식** (모델 아키텍처 수정, 추가 데이터 수집)은 기존에 존재하는 모델을 활용하는 측면에서 효율적이지 않음

- 본 논문에서는 LM의 output과 생성 프로세스의 가이드 역할을 하는 text critic classifier의 output을 결합하는 **Critic-driven decoding approach**를 제안함

- LM의 아키텍처를 변경할 필요가 없고 추가적인 데이터의 학습이 필요하지 않기 때문에 기존에 존재하는 모델에 쉽게 결합 가능함

Critic-driven decoding

- **Data-to-Text generation**

$$P(y|x) = \prod_{i=1}^{n} P(y_i|y_{\leq i-1}, x)$$

- **Additional Text Generation Critic c**

$$P(y|x,c) = \prod_{i=1}^{n} P(y_i|y_{\leq i-1}, x, c)$$

- 생성된 텍스트와 입력 데이터 representation 간의 match를 평가함
- Critic: binary variable -> 텍스트가 입력 텍스트와 일치하면 1, 아니면 0

$$P(y_i|y_{\leq i-1}, x, c) \propto P(c|y_{\leq i}, x)P(y_i|y_{\leq i-1}, x)$$

Training
a text generation critic

- 텍스트 생성 시 생성된 텍스트가 입력 데이터와 일치하는 지를 측정함
- Backbone: LM의 인코더
- **negative example은 다음의 다섯 가지 방법을 사용하여 생성함**
    1. **base**: 각각의 positive example에 대해 마지막 토큰을 랜덤 토큰으로 대체 / train set을 더욱 어렵게 구성하기 위해 토큰은 동일한 데이터에 대한 다른 텍스트나 데이터셋의 다른 랜덤 문장에서 샘플링함
    2. **base with full sentences**: 데이터셋의 랜덤 문장으로 대체
    3. **vanilla LM**: LM을 통해 가장 다음 토큰의 가능성이 높은 5개의 토큰에서 랜덤으로 토큰을 선택하여 구성함
    4. **fine-tuned LM**: 해당 벤치마크를 fine-tuning한 모델을 사용하여 랜덤 토큰을 선택함
    5. **fine-tuned LM with full sentences**

Experimental evaluation

Classification performance

critic model	accuracy	F1
1. base	0.969	0.970
2. base w/full sent.	0.984	0.975
3. vanilla. LM	0.931	0.798
4. fine-tuned LM	0.920	0.718
5. fine-tuned LM w/full sent.	0.929	0.714

Experimental evaluation

Results of automatic evaluation on the WebNLG test set

decoding approach	BLEU	METEOR	BERT Score	NLI			BLEURT		
				all	ood	ind	all	ood	ind
baseline	45.09	0.373	0.911	0.841	0.783	0.889	0.128	-0.026	0.257
1. critic (base)	45.48	**0.377**	**0.913**	0.855	0.801	0.901	**0.155**	**0.010**	**0.277**
2. critic (base with full sentences)	44.90	0.371	**0.913**	**0.868**	**0.820**	**0.909**	0.153	0.007	0.274
3. critic (vanilla LM)	45.44	**0.377**	**0.913**	0.859	0.811	0.900	0.139	-0.002	0.258
4. critic (fine-tuned LM)	45.41	0.373	0.911	0.834	0.772	0.886	0.128	-0.021	0.254
5. critic (fine-tuned LM w. full sentences)	**45.59**	0.374	0.912	0.839	0.779	0.889	0.136	-0.013	0.261

Results of automatic evaluation on the OpenDialKG test set

	BLEU	METEOR	BERTScore	NLI	BLEURT
baseline	11.74	0.149	0.775	0.748	-0.933
1. critic (base)	9.67	0.137	0.771	**0.796**	**-0.905**
2. critic (base with full sentences)	**11.88**	**0.151**	**0.776**	0.754	-0.920
3. critic (vanilla LM)	10.37	0.139	0.763	0.713	-0.980
4. critic (fine-tuned LM)	10.76	0.143	0.768	0.739	-0.964
5. critic (fine-tuned LM with full sentences)	11.41	0.149	0.771	0.712	-0.956

Experimental evaluation

Analysis of introduced changes

critic model	mod [%]	add.	rem.
base	66.3	4.54	4.58
base w/full sent.	72.8	5.42	4.72
vanilla LM	72.8	5.03	5.39
fine-tuned LM	48.5	2.52	2.71
fine-tuned LM w/full sent.	31.9	1.63	1.76

Results of manual evaluation

decoding approach	min. hal.	maj. hal.	omi.	disfl.	rep.	avg. rank
baseline	0.22	0.40	0.25	0.20	0.08	3.61
1. critic (base)	0.21	0.30	**0.20**	0.17	**0.04**	**3.38**
2. critic (base with full sentences)	0.21	**0.29**	0.27	**0.11**	0.08	3.43
3. critic (vanilla LM)	**0.18**	**0.29**	0.23	0.19	0.05	3.54
4. critic (fine-tuned LM)	0.22	0.37	0.26	0.21	0.07	3.53
5. critic (fine-tuned LM with full sentences)	0.20	0.37	0.26	0.18	0.07	3.54

Conclusion

- Data-to-Text generation 작업에서 hallucination을 완화하기 위한 새로운 critic-driven decoding approach를 소개함
- Classifier의 output을 사용하여 기존의 LM을 수정하지 않고도 생성 프로세스에 개입할 수 있음
- WebNLG과 OpenDialKG의 실험 결과는 제안된 방법이 텍스트 생성 성능을 방해하지 않고 hallucination을 완화할 가능성이 있음을 보여줌

Thank You