组会

2020.12.12 许晓丹

PAIR: Planning and Iterative Refinement in Pre-trained Transformers for Long Text Generation

Xinyu Hua
Khoury College of Computer Sciences
Northeastern University
Boston, MA
hua.x@northeastern.edu

Lu Wang
Computer Science and Engineering
University of Michigan
Ann Arbor, MI
wangluxy@umich.edu

Motivation

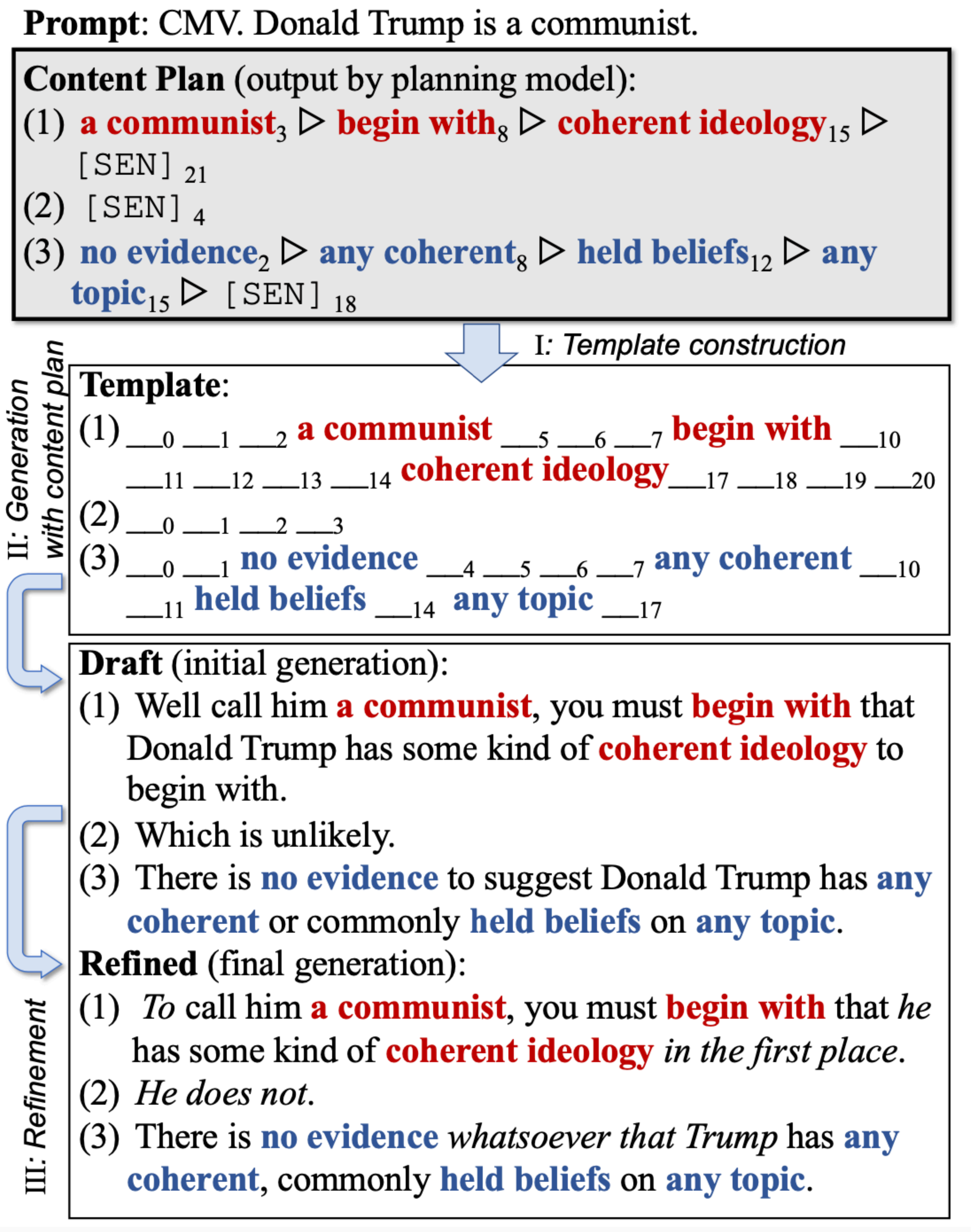
1. 虽然GPT-2等可以产生plausible text，但是用户无法指定要包含的内容和顺序；
2. 告诉大模型 *when to say what*可以提高它的实用性；
3. 现有的content plan模型都需要模型修改和重新训练，代价非常昂贵；
4. this work aims to bring new insights into *how to effectively incorporate content plans into larg models to generate more relevant and coherent text*.

Work

1. Propose a content planner based on BERT;
2. Propose a content-controlled text generation framework based BART;
3. present an iterative refinement algorithm.

DataSet

1. 对抗性观点生成 ——Reddit ChangeMyView;
2. 文章观点生成——NYT;
3. 新闻报道生成——NYT。



Content Planning with BERT

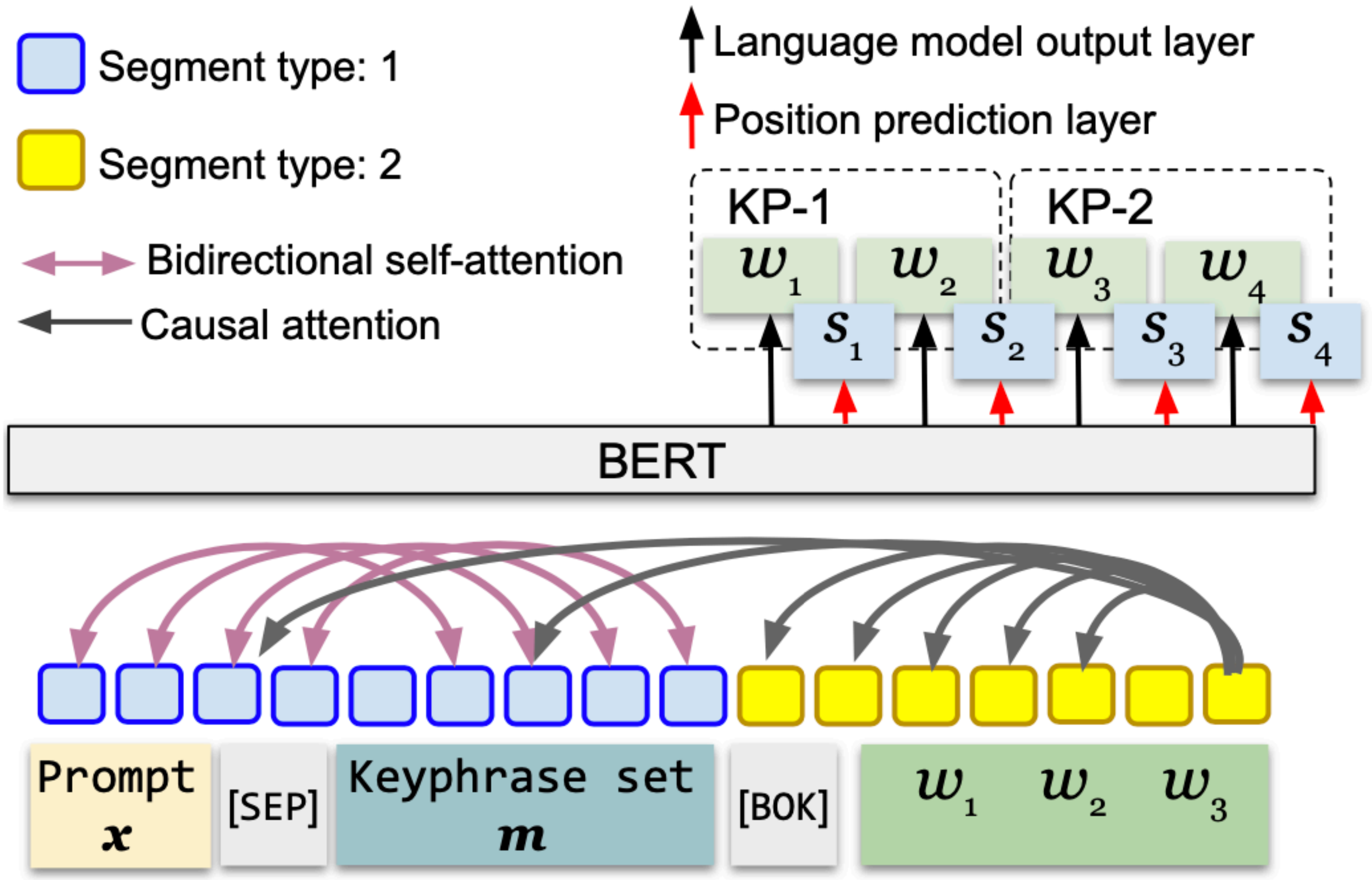


Figure 2: Content planning with BERT. We use bidirectional self-attentions for input encoding, and apply causal self-attentions for keyphrase assignment and position prediction. The input $(\boldsymbol{x}, \boldsymbol{m})$ and output keyphrase assignments $(\boldsymbol{m}')$ are distinguished by different segment embeddings.

输入： Prompt x +a set of keyphrases m that are relevant to the prompt;

输出： Keyphrase assignments + positions

a communist begin with coherent ideology [SEN] [...] + *positions*

Iterative Refinement

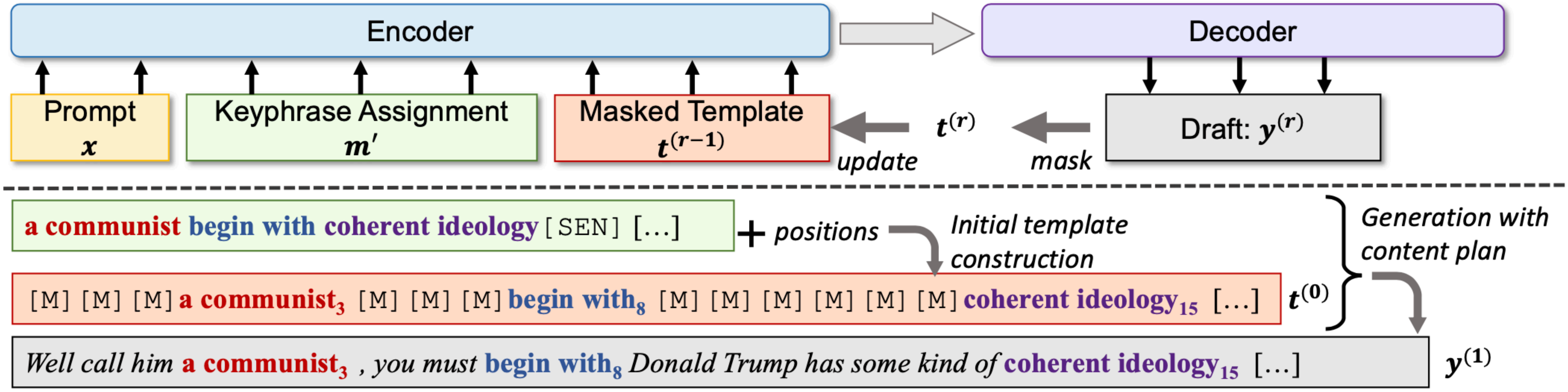


Figure 3: Our content-controlled text generation framework, PAIR, which is built on BART. Decoding is executed iteratively. At each iteration, the encoder consumes the input prompt $\boldsymbol{x}$, the keyphrase assignments $\boldsymbol{m}'$, as well as a partially masked template ($\boldsymbol{t}^{(r-1)}$ for the r-th iteration, `[M]` for masks). The autoregressive decoder produces a complete sequence $\boldsymbol{y}^{(r)}$, a subset of which is further masked, to serve as the next iteration's template $\boldsymbol{t}^{(r)}$.

Iterative Refinement

做法:

At each iteration, the n least confident tokens are replaced with [MASK];

实验：with ground truth content plan

	ArgGen				Opinion				News			
	B-4	**R-L**	**MTR**	**Len.**	**B-4**	**R-L**	**MTR**	**Len.**	**B-4**	**R-L**	**MTR**	**Len.**
Seq2seq	0.76	13.80	9.36	97	1.42	15.97	10.97	156	1.11	15.60	10.10	242
KPSeq2seq	6.78	19.43	15.98	97	11.38	22.75	18.38	164	11.61	21.05	18.61	286
$\text{PAIR}_{\text{light}}$	26.38	47.97	31.64	119	16.27	33.30	24.32	210	28.03	43.39	27.70	272
$\text{PAIR}_{\text{light}}$ w/o refine	25.17	46.84	31.31	120	15.45	32.35	24.11	214	27.32	43.08	27.35	278
$\text{PAIR}_{\text{full}}$	**36.09**	**56.86**	**33.30**	102	**23.12**	**40.53**	**24.73**	167	**34.37**	**51.10**	**29.50**	259
$\text{PAIR}_{\text{full}}$ w/o refine	34.09	55.42	32.74	101	22.17	39.71	24.65	169	33.48	50.27	29.26	260

Table 2: Key results on argument generation, opinion article writing, and news report generation. BLEU-4 (B-4), ROUGE-L (R-L), METEOR (MTR), and average output lengths are reported (for references, the lengths are 100, 166, and 250, respectively). $\text{PAIR}_{\text{light}}$, using keyphrase assignments only, consistently outperforms baselines; adding keyphrase positions, $\text{PAIR}_{\text{full}}$ further boosts scores. Improvements by our models over baselines are all significant ($p < 0.0001$, approximate randomization test). Iterative refinement helps on both setups.

实验：with ground truth content plan

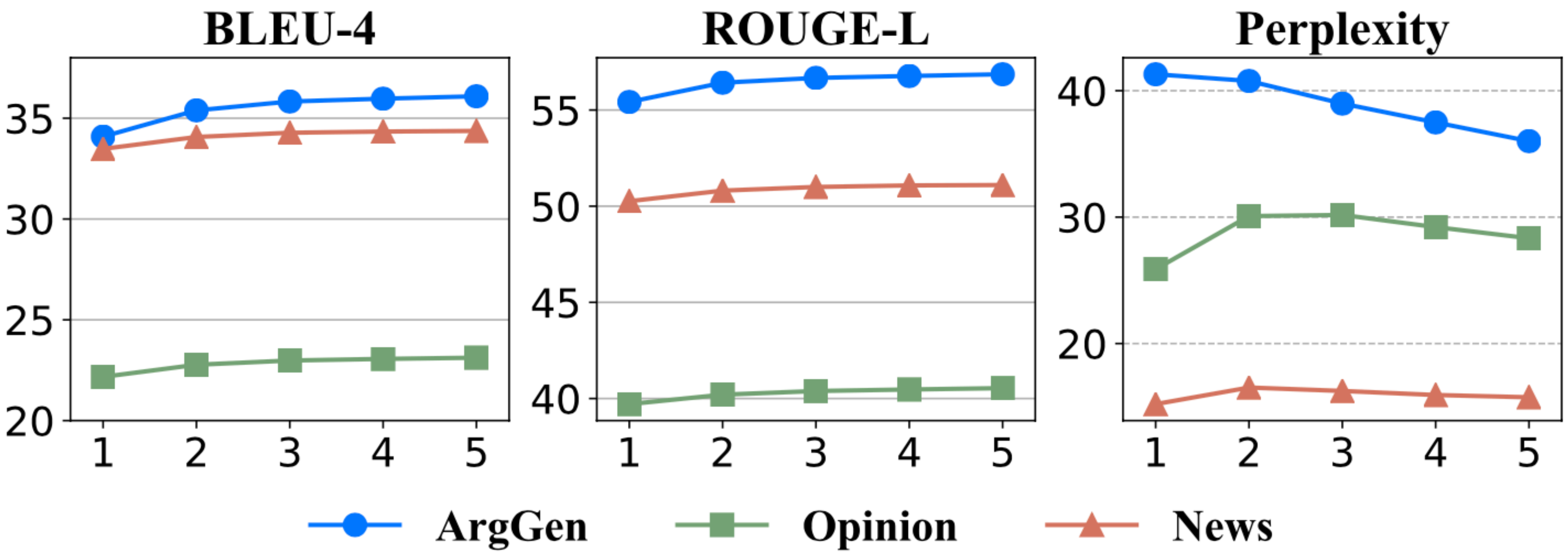


Figure 4: Results on iterative refinement with five iterations. Both BLEU and ROUGE-L scores steadily increase, with perplexity lowers in later iterations.

实验：with Predicted Content Plans

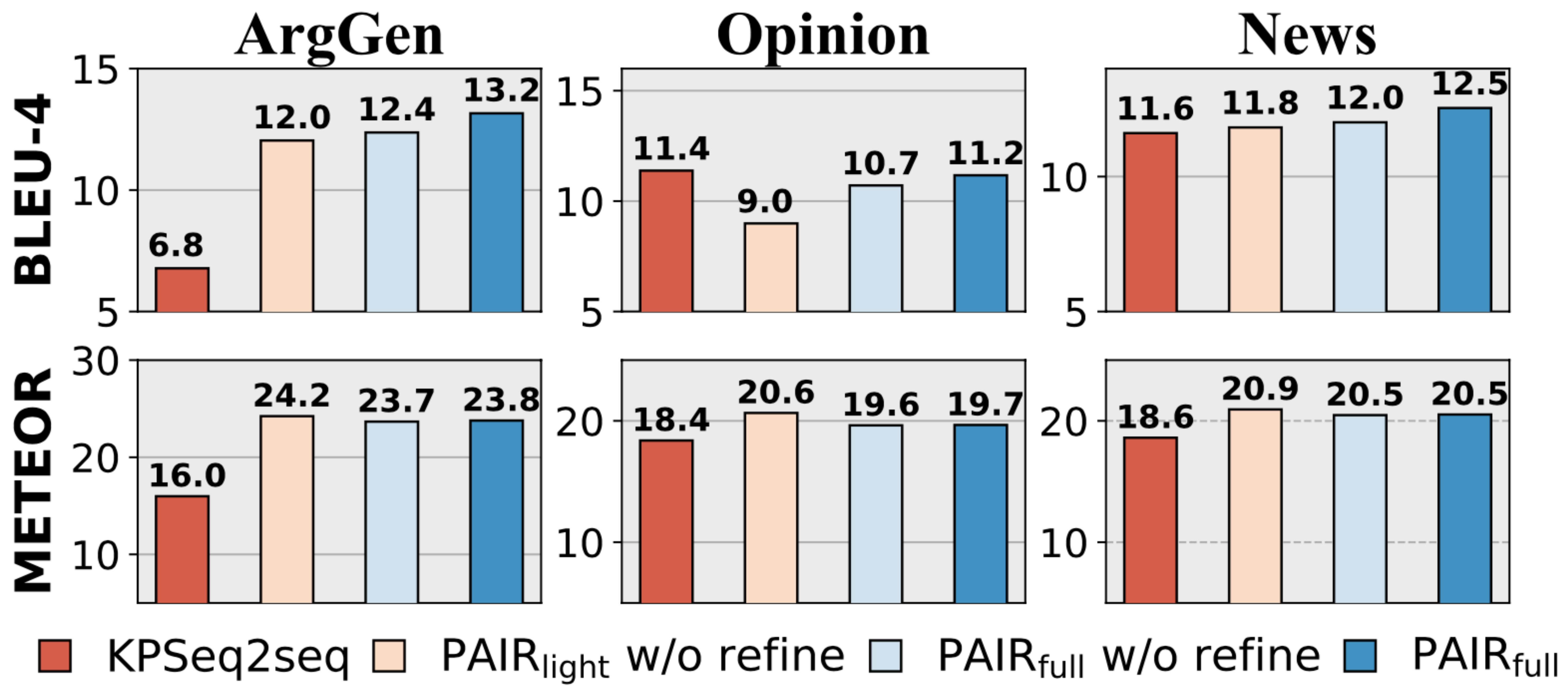


Figure 5: End-to-end generation results with automatically predicted content plans. Our models outperform KPSEQ2SEQ in both metrics, except for BLEU-4 on opinion articles where results are comparable.

Human Evaluation

ArgGen	**Fluency**	**Coherence**	**Relevance**
KPSeq2seq	4.63	3.28	2.79
$\text{PAIR}_{\text{light}}$	**4.75**	**3.97***	**3.85***
$\text{PAIR}_{\text{full}}$	4.46	3.76*	3.79*

Table 3: Human evaluation for argument generation on fluency, coherence, and relevance, with 5 as the best. The Krippendorff's α are 0.28, 0.30, and 0.37, respectively. Our model outputs are significantly more coherent and relevant than KPSeq2seq (*: $p < 0.0001$), with comparable fluency.

ENT-DESC: Entity Description Generation by Exploring Knowledge Graph

Liying Cheng[*1,2], **Dekun Wu**[†3], **Lidong Bing**[2], **Yan Zhang**[†1], **Zhanming Jie**[†1], **Wei Lu**[1], **Luo Si**[2]
[1] Singapore University of Technology and Design
[2] DAMO Academy, Alibaba Group [3] York University, Canada
{liying.cheng, l.bing, luo.si}@alibaba-inc.com, jackwu@eecs.yorku.ca,
{yan_zhang, zhanming_jie}@mymail.sutd.edu.sg, luwei@sutd.edu.sg

Motivation

1. 应用场景：KG-to-text——给定main entity和这个entity周围的relations，输出描述；
2. 在现有的数据集中（比如WIKIBIO、webNLG等），输出和输入的triple有良好的对齐方式。但实际上，输入的信息会有冗余。也就是说，输出可能只会覆盖最重要的信息。

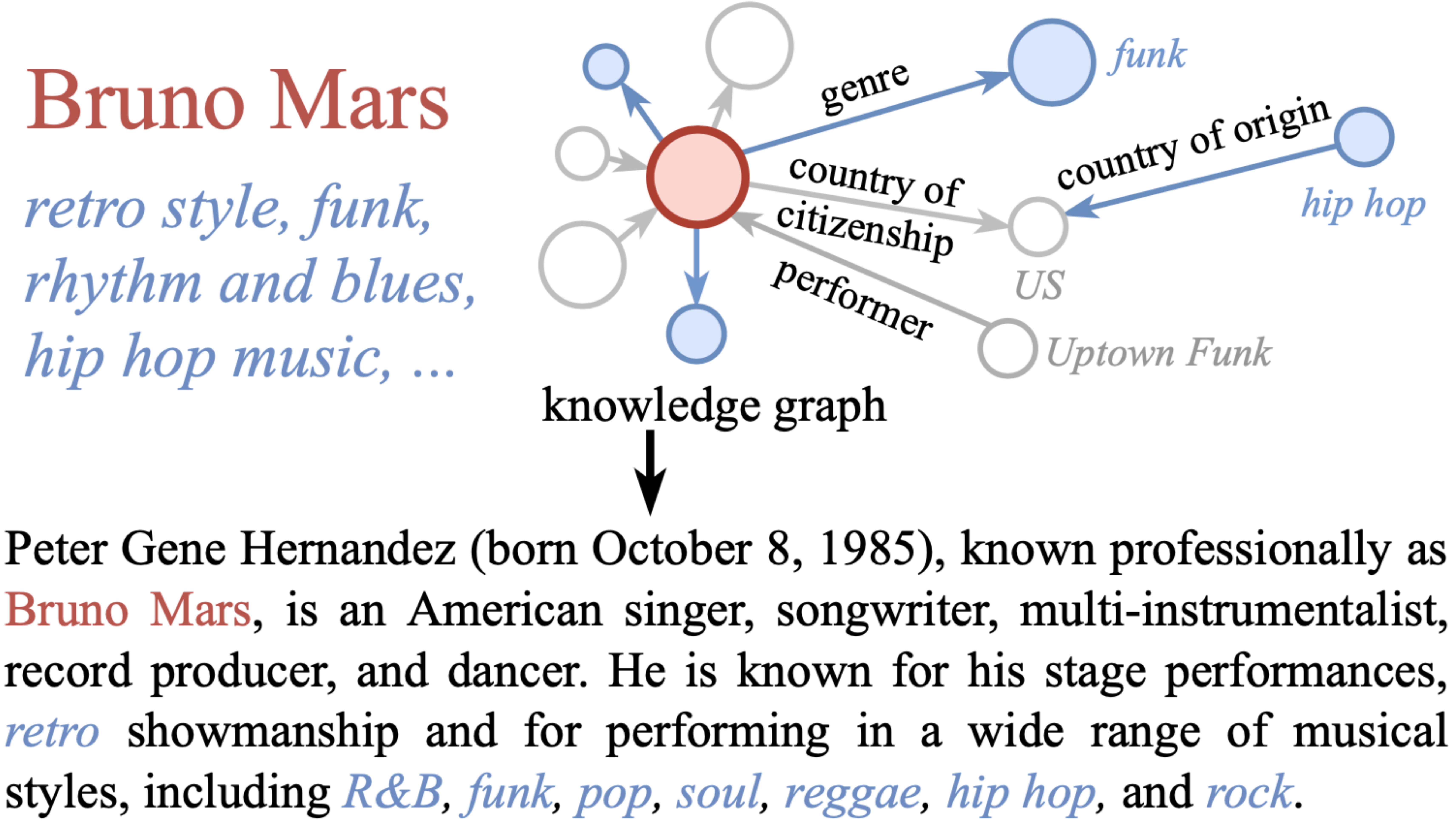


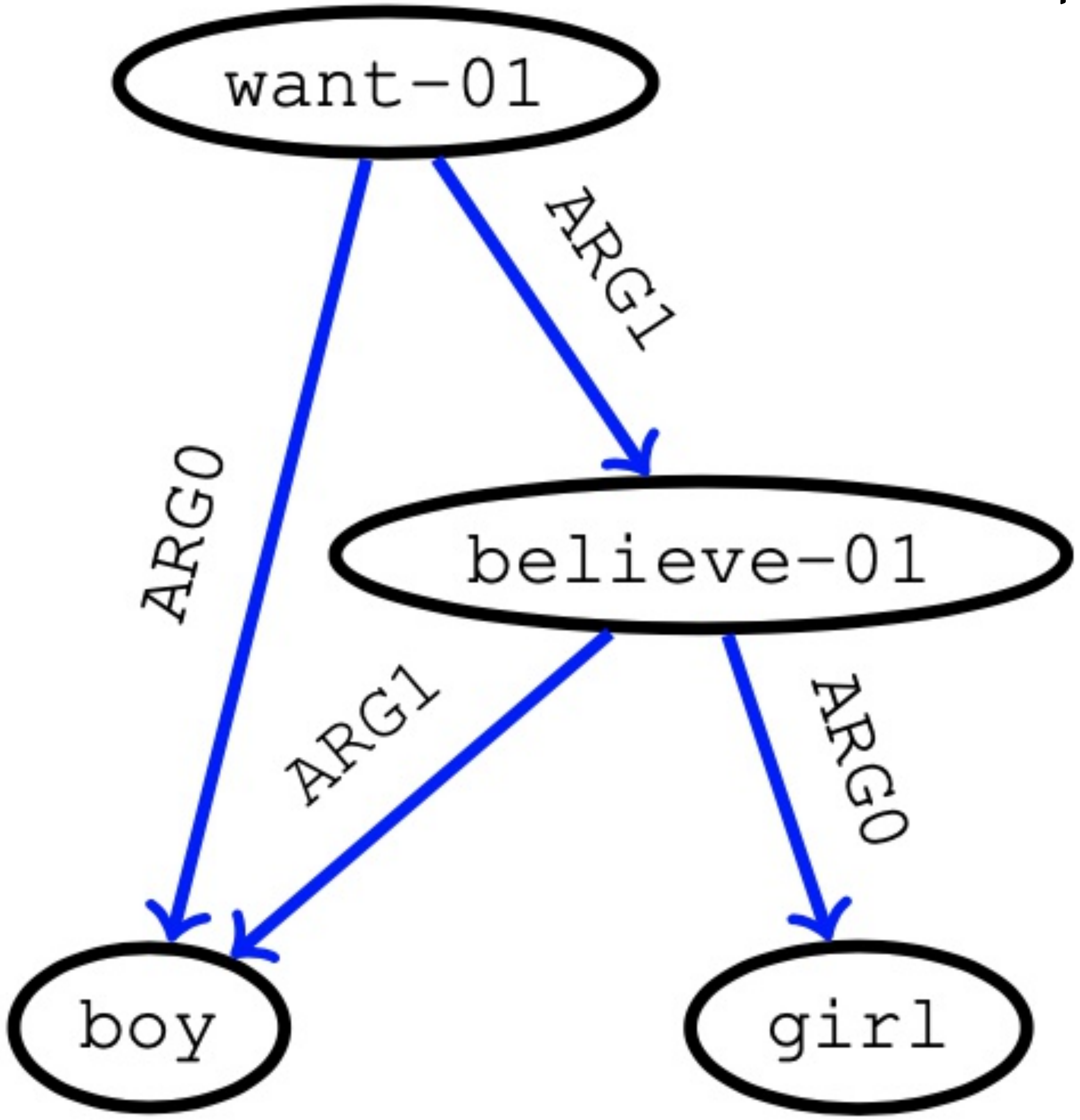
Figure 1: An example showing our proposed task.

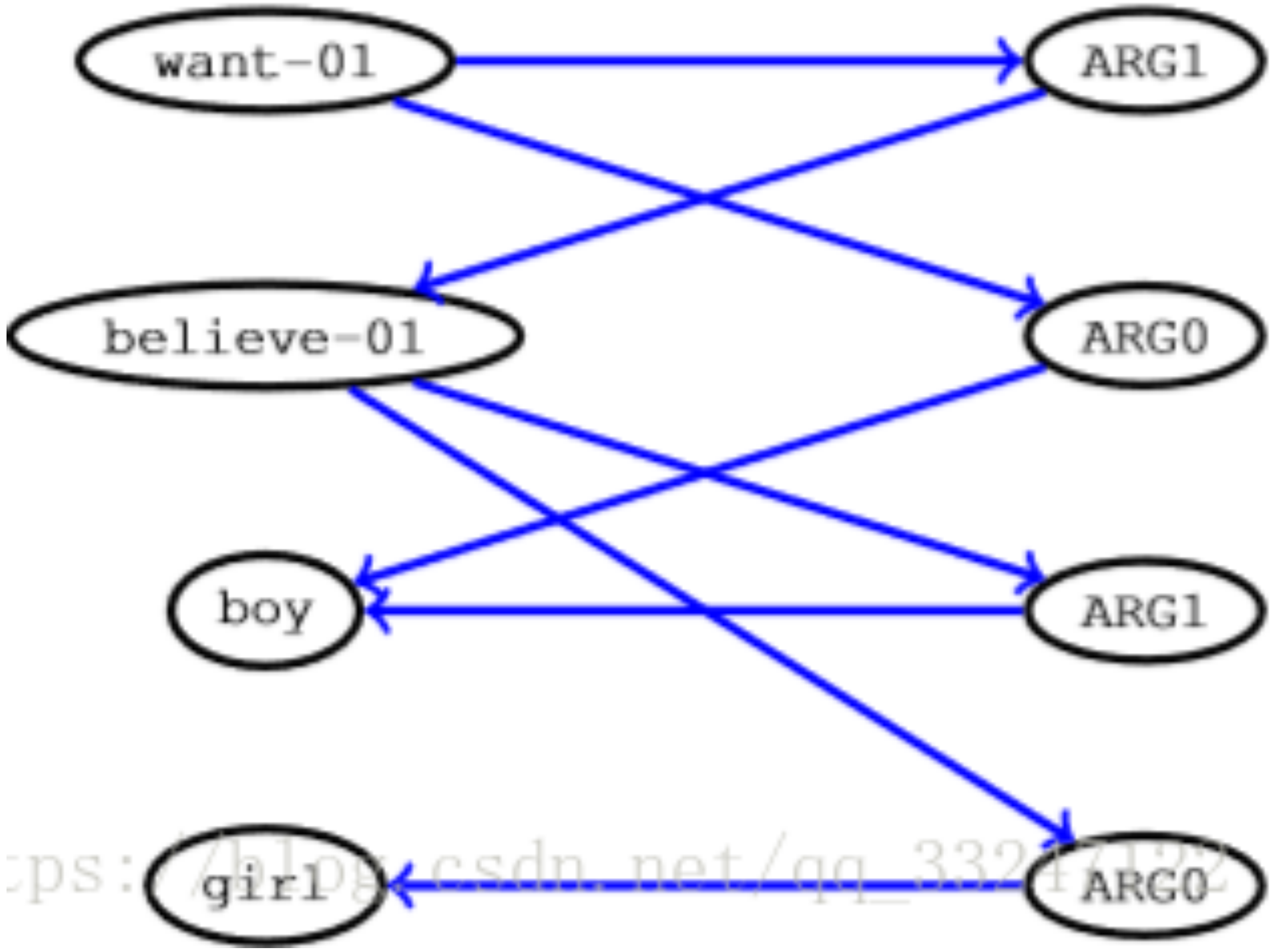
Work

1. 提出了一个KG-to-text的数据集：ENT-DESC；在输出和输入之间缺少显式的对齐关系；
2. 提出了multi-graph transformation + aggregation layer.

前人做法：Levi graph

有缺点





Model： multi-graph transformation

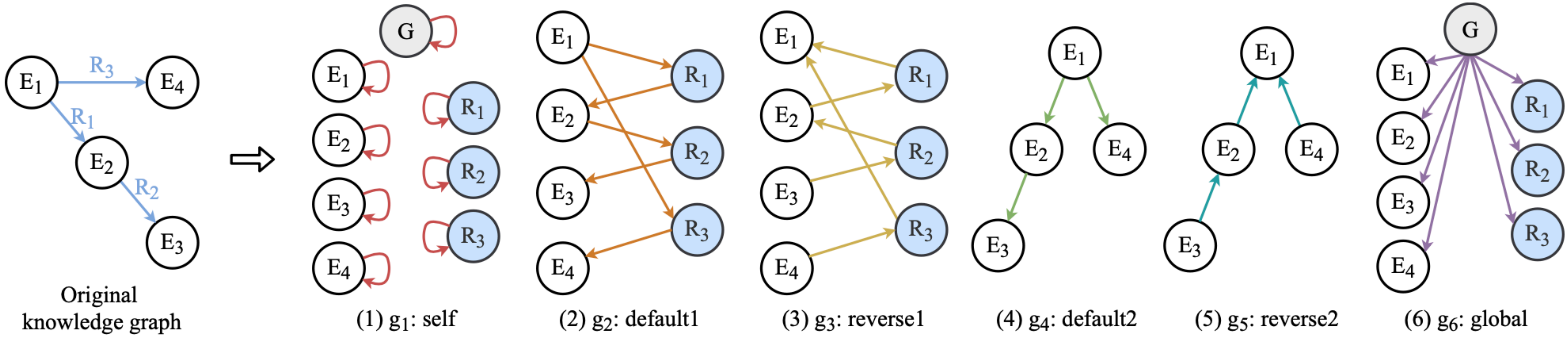


Figure 4: An example of multi-graph transformation.

Model

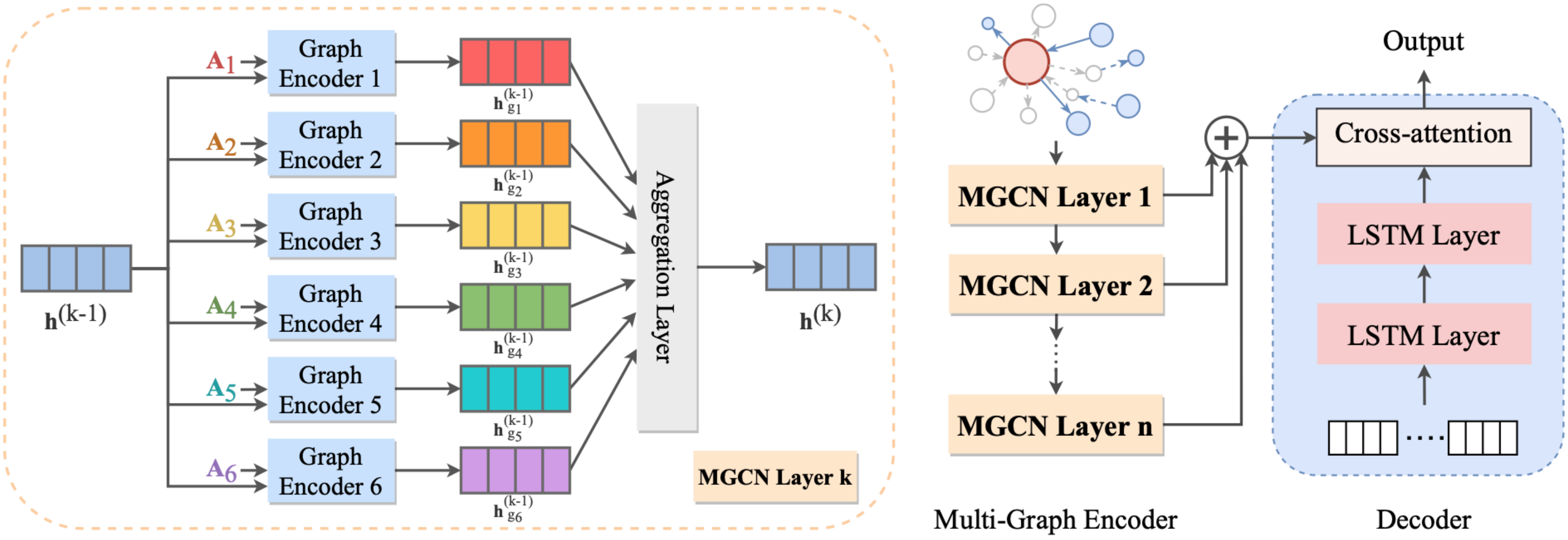


Figure 3: Overview of our model architecture. There are n MGCN layers in the multi-graph encoder, and 2 LSTM layers in the decoder. $\mathbf{h}^{(k-1)}$ is the input graph representation at Layer k, and its 6 copies together with the corresponding adjacent matrices $\mathbf{A}_i$'s of transformed graphs in the multi graph (refer to Figure 4) are fed into individual basic encoders. Finally, we obtain the graph representation $\mathbf{h}^{(k)}$ for the next layer by aggregating the representations from these encoders.

Model：aggregation layer

1. Sum-based;
2. Average based;
3. CNN based.

实验

Models	BLEU	METEOR	TER↓	ROUGE$_1$	ROUGE$_2$	ROUGE$_L$	PARENT
S2S (Bahdanau et al., 2014)	6.8	10.8	80.9	38.1	21.5	40.7	10.0
GraphTransformer (Koncel-Kedziorski et al., 2019)	19.1	16.1	94.5	53.7	37.6	54.3	21.4
GRN (Beck et al., 2018)	24.4	18.9	70.8	54.1	38.3	55.5	21.3
GCN (Marcheggiani and Perez-Beltrachini, 2018)	24.8	19.3	70.4	54.9	39.1	56.2	21.8
DeepGCN (Guo et al., 2019)	24.9	19.3	70.2	55.0	39.3	56.2	21.8
MGCN	**25.7**	**19.8**	**69.3**	**55.8**	**40.0**	**57.0**	**23.5**
MGCN + CNN	**26.4**	**20.4**	69.4	56.4	40.5	**57.4**	**24.2**
MGCN + AVG	26.1	20.2	**69.2**	56.4	40.3	57.3	23.9
MGCN + SUM	**26.4**	20.3	69.8	56.4	**40.6**	**57.4**	23.9
GCN + delex	28.4	22.9	65.9	61.8	45.5	62.1	30.2
MGCN + CNN + delex	29.6	**23.7**	**63.2**	**63.0**	**46.7**	**63.2**	**31.9**
MGCN + SUM + delex	**30.0**	**23.7**	67.4	62.6	46.3	62.7	31.5
The rows below are results of generating from entities only without exploring the KG.							
E2S	23.3	20.4	68.7	58.8	41.9	58.2	27.7
E2S + delex	21.8	20.5	67.5	59.5	39.5	59.2	23.4
E2S-MEF	24.2	21.3	65.8	59.8	43.3	60.0	26.3
E2S-MEF + delex	20.6	20.3	66.5	59.1	40.0	59.3	24.3

Table 2: Main results of models on ENT-DESC dataset. ↓ indicates lower is better.

Ablation Study

Model	**BLEU**	**Δ (BLEU)**
MGCN + SUM	26.4	-
– g_6: *global*	26.0	-0.4
– g_5: *reverse2*	25.8	-0.6
– g_4: *default2*	26.1	-0.3
– g_3: *reverse1*	25.7	-0.7
– g_2: *default1*	26.1	-0.3
MGCN	25.7	-0.7
GCN	24.8	-1.4

Table 4: Results of the ablation study.

Thanks