Pretraining Language Model through Text and Knowledge Graph Loop with Reconstruction Error

Anonymous ACL submission

Abstract

Humans encode natural language (NL) inputs into knowledge graphs (KG), and conversely, decode knowledge graphs into natural language outputs. For instance, the statement, *"New York is one of the most crowded cities in America,"* can be distilled into entity-relation knowledge as *(New York, located in, America)* and *(New York, has, large population)*. Extensive research has been conducted on the interrelationship between NL and KG, focusing on either synergistic frameworks or translations from one to the other. In this study, we propose a novel pretraining approach that conceptualizes NL-KG-NL as **an unsupervised sequential loop** (see in Figure 1) rather than a single lane, akin to human information processing. Specifically, a generative model is designed to perform three functions: 1) extracting a knowledge graph from natural language (encoding), 2) verbalizing a knowledge graph to natural language (decoding), that forms a continuous and coherent loop, and 3) recovering the knowledge graph from incrementally masked tokens (memorizing). During the unsupervised training phase, the model aims to minimize two reconstruction errors through the NL-KG-NL loop and masked KG. With the proposed approach, the model 1) clearly exposes an interpretable intermediate stage in pre-training; 2) acquires extra attention on factual and relational knowledge; 3) requires no text annotation, suitable for low-resource, customized fields.

1 Introduction

Pretraining large language models (LLMs) on unsupervised tasks, such as masked token prediction and next token prediction, has demonstrated remarkable performance across various downstream tasks, including natural language understanding and reasoning (Achiam et al., 2023; Raffel et al., 2020; Cheng et al., 2023; Sharma et al., 2022; Liu et al., 2023). This unsupervised pretraining on web-scale text allows the language model to effectively capture surface-level token correlations, for example, learning to predict sequences like *open the door* rather than *open the pencil*. Despite acquiring extensive world knowledge from training texts, the model's black-box nature remains a significant challenge for researchers seeking to interpret and improve on the downstream tasks. Numerous studies have analyzed attention mechanisms (Hewitt et al., 2023; Von Oswald et al., 2023; Arora and Goyal, 2023) and neurosymbolic methodologies (Singh et al., 2023; Liu et al., 2023; Zhang et al., 2023) to address these issues. In contrast, our work proposes a novel pretraining framework — encoding-memorizing-decoding — designed to imitate human cognitive processes, therefore, enhancing the interpretability and controllability of these LLMs.

An array of work has explored pretraining tasks in encoder-decoder language models, including the BART (Lewis et al., 2020) and T5 (Raffel et al., 2020; Chung et al., 2024; Tay et al., 2022) families. These models are pretrained using unsupervised tasks such as masked token prediction and next token prediction. While these pretraining tasks prove beneficial for downstream tasks like question answering, translation, and summarization, researchers continue to face challenges in explaining how these pretraining tasks facilitate the acquisition of world knowledge by the models. Furthermore, the two dominant unsupervised pretraining tasks do not fully capture the information embedded in text, such as entities and their relationships, leading to a suboptimal learning process. To address this, we propose a novel unsupervised pretraining framework that captures deeper relationships among entities, drawing inspiration from human learning.

Humans learn by distilling new knowledge from textual inputs and integrating it into their mental models (Piaget, 1952). For example, given the sentence "*New York is one of the most crowded cities*

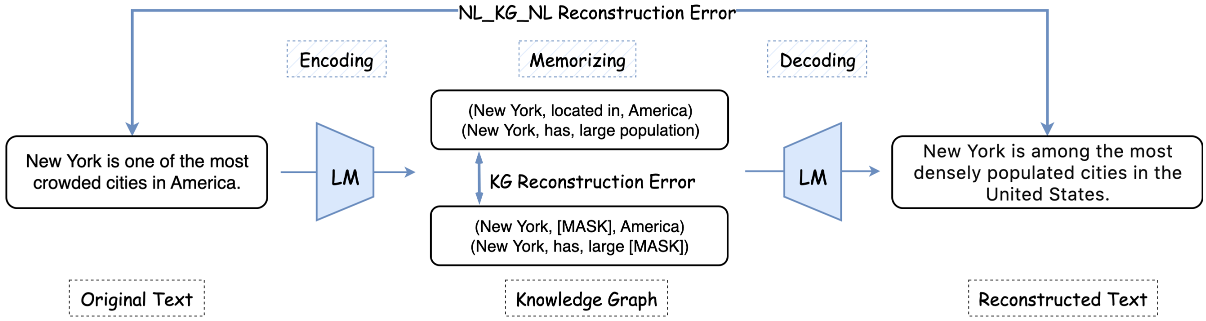

Figure 1: An example of the NL-KG-NL loop to train a generative model on reconstruction errors. We propose an encoding-memorizing-decoding pretraining framework that mimics human cognitive process. For information encoding and decoding, the LM learns to extract graph-based knowledge from textual data and then generates reconstructed text from the knowledge graph. The objective is to minimize the reconstruction error between the original text and the reconstructed text. For information memorization, the LM trains using incremental masking on the entities and relations to accurately reconstruct the original knowledge graph.

in America," a human might first extract two key pieces of information: (*New York, located in, America*) and (*New York, has, large population*). Subsequently, this new information can be integrated into their existing internal knowledge base, which might already include (*New York, is, city*) and (*America, is, country*), resulting in a refined understanding: (*city – New York, located in, country – America*) and (*New York, has, large population*). Later, a human could express this knowledge by constructing a sentence such as "*New York is among the most densely populated cities in the United States.*" Our brains process information in a manner like an hourglass: during encoding, unnecessary signals are filtered out, with core components stored as knowledge graphs; during decoding, expressive formats are added back for communication. In contrast, current large language models operate by predicting the next token in a sequence, copying and pasting natural language text without considering this hierarchical process as in the human brain. As a result, the model's outputs can be challenging for humans to interpret and control.

Inspired by human learning, we propose a novel pretraining task for language models that mimics the hierarchical process of human information processing, as illustrated in Figure 1. Our pretraining framework consists of three stages: encoding, memorizing, and decoding, each reflecting a fundamental cognitive skill of the human brain.

Information Encoding: The language model is trained to extract knowledge graphs from natural language text (left).

Information Memorizing: The model attempts to recover masked knowledge, such as entities and relations, within the knowledge graph (middle).

Information Decoding: The model verbalizes the knowledge graph into natural language outputs (right).

Following this procedure, the language model is optimized unsupervisedly through two types of reconstruction errors: NL-KG-NL reconstruction error during the encoding-decoding phase, and masked token reconstruction error during the memorizing phase. We evaluate the model's performance across various downstream tasks, including natural language inference (NLI) and question answering (QA).

Unlike the standard next-token prediction training task, our method fully leverages the data through three distinct tasks: encoding as a knowledge graph (KG), memorizing the KG, and decoding it back into natural language (NL). The evaluation results show…

In summary, our contributions are two-fold:

1. We propose a novel pretraining framework: encoding-memorizing-decoding.

2. We evaluate this framework on various downstream tasks.

2 Method

This innovative methodology mimics the information processing of the human brain. For information encoding and decoding, the language model learns to extract graph-based knowledge from textual data and then generates reconstructed text from the knowledge graph. The objective is to minimize the reconstruction error between the original text and the reconstructed text. A KL divergence is added at each step, encoding (forming a knowledge graph) and decoding (generating the reconstructed text), to avoid catastrophic forgetting in the pretraining stage. For information memorization, the language model trains using incremental masking on entities and relations to accurately reconstruct the original knowledge graph.

Encoding: Initially, we incorporate natural language text and instruct the language model to extract knowledge graphs in the form of (subject, relation, object) tuples. The natural language text , concatenated after the natural language-to-knowledge graph prompt (denoted as $NL2KGPrompt + Text$), as input to the encoder, while the decoder generates the knowledge graph as output. Since this process involves unsupervised learning, no gold-standard annotations are required for the extracted knowledge graphs.

Decoding: We input only the knowledge graph generated during the encoding phase into the language model and instruct it to produce a coherent textual output. Specifically, we concatenate the knowledge graph into the prompt, referred to as $KG2NL\_Prompt + KG$. The decoder then generates the reconstructed text as the output. Ideally, the generated output should match the original input in both syntax and semantics, thereby forming a closed unsupervised loop. In practice, we calculate the token-level cross-entropy loss between the original and reconstructed texts, which serves as the reconstruction error ($L_{NL\_KG\_NL}$). To optimize the two-layer encoder-decoder model, we use aggregated embeddings (logits*embeddings) instead of argmax logits embeddings as input during the decoding phase.

Memorization: The knowledge graph is incrementally masked by randomly selecting tokens, including both entities and relations. The model is then tasked with predicting the masked tokens, and the cross-entropy loss of these predictions, denoted as L_{KG}, is calculated. The same encoder-decoder model is used here.

Our objective can be defined as:

$$L(X) = L_{NL\_KG\_NL}(X_{original}, X_{reconstructed})$$
$$+ L_{KG}(X_{KG}, X_{masked\_KG})$$
$$+KL_{NL}(LM_{orig}X_{reconstructed}, LM_{trained}X_{reconstructed})$$
$$+ KL_{KG}(LM_{orig}X_{KG}, LM_{trained}X_{KG})$$

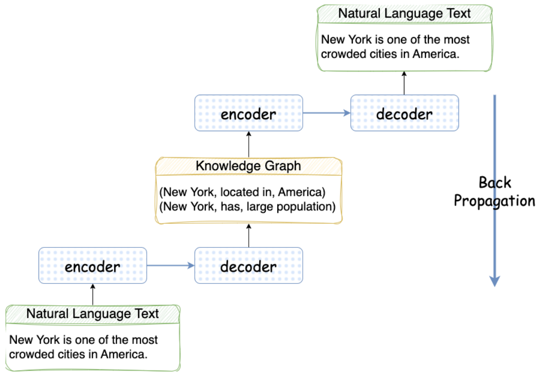

Figure 2: An example of the Encoder-Decoder model architecture for pretraining NL-KG-NL loop.

where $L_{NL\_KG\_NL}$ indicates the reconstruction error between the input text and the output text, and L_{KG} represents the reconstruction error between the original knowledge graph and the masked knowledge graph.

3 Implementation

The NL-KG-NL loop pretraining task is implemented on both encoder-decoder architecture and decoder-only architecture with slightly different setup.

The encoder-decoder language model is shared across the encoding, decoding, and memorization phases 1. In the encoding phase, the model extracts knowledge graphs from natural language text. During the decoding phase, it is tasked with verbalizing the knowledge graph back into natural language text. Reconstruction errors are backpropagated throughout the encoding-decoding phases. The same language model is also employed in the memorization phase to reconstruct masked tokens.

The decoder-only language model takes as input the natural language text and generates the knowledge graphs in the encoding phase, and vise versa in the decoding phase. Reconstruction errors are backpropagated throughout the encoding-decoding phases only on the inputs part.

4 Data

Our natural language pretraining data include dataset available on the web, such as Wikipedia, Wikidata or and dataset that is suitable for extracting knowledge graphs, e.g., HaulEval.

we evaluated on a diversity of validation dataset and held out dataset. The validation dataset incldu-ing SQuAD, SQuAD2.0, HaluEval<doc, sum>.

5 Experiments

We explore an adapter method, such as LoRA, Adapter Fusion, for efficient pretraining on LLMs.

	HaluEval <doc, sum>	SQuAD
encoder-decoder model		
flan-t5-xl	26 / 31	90.3 / 91.3
decoder model	Row 3, Col 2	Row 3, Col 3
llama	Row 3, Col 2	Row 3, Col 3

Table 1: Evaluation Results on downstream tasks

6 Results

7 Related Work

Knowledge Graph Application: The interrelationship between textual data and Knowledge Graphs (KGs) has been extensively explored by researchers across various subfields. One such area involves the construction of KGs from natural language (NL) texts (Pan et al., 2024; Kumar et al., 2020), while another focuses on generating coherent NL texts from KGs (Pan et al., 2024; Ke et al., 2021). A third area examines the synergistic integration of both KGs and NL texts in training language models (LMs) (Shen et al., 2020; Sun et al., 2021; Yu et al., 2022; Yasunaga et al., 2022). However, irrespective of the approach, all methods necessitate a high-quality, KG-text aligned corpus, which is expensive to obtain. Our approach eliminates this requirement and facilitates the model training by tackling the reconstruction error in either format (KG or NL text), while designing the reconstruction loop incorporating both formats.

Language Model Pretraining: Today's LLMs are trained on the task of next token prediction, P(xn|x1...xn-1), rendering them susceptible to producing hallucinations (Pan et al., 2024). In contrast, our approach goes beyond mere token-level conditional prediction by enhancing LLMs through knowledge-level condition generation. Specifically, the model is capable of extracting structured knowledge from a given passage, represented as P(KG|NL), while express in natural language given a knowledge graph, formulated as P(NL|KG). Another emerging training technique for language models is latent diffusion, in which natural language input is incrementally transformed into random noise and subsequently reconstructed (Rombach et al., 2022). Compared to this architecture, our method converts NL into a KG with perturbations and reconstructs the NL from KG then.

8 Experimental Results and Analysis

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Sanjeev Arora and Anirudh Goyal. 2023. A theory for emergence of complex skills in language models. *arXiv preprint arXiv:2307.15936*.

Zhoujun Cheng, Tianbao Xie, Peng Shi, Chengzu Li, Rahul Nadkarni, Yushi Hu, Caiming Xiong, Dragomir Radev, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Tao Yu. 2023. Binding language models in symbolic languages. In *The Eleventh International Conference on Learning Representations*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

John Hewitt, John Thickstun, Christopher Manning, and Percy Liang. 2023. Backpack language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9103–9125, Toronto, Canada. Association for Computational Linguistics.

Pei Ke, Haozhe Ji, Yu Ran, Xin Cui, Liwei Wang, Linfeng Song, Xiaoyan Zhu, and Minlie Huang. 2021. Jointgt: Graph-text joint representation learning for text generation from knowledge graphs. *arXiv preprint arXiv:2106.10502*.

Abhijeet Kumar, Abhishek Pandey, Rohit Gadia, and Mridul Mishra. 2020. Building knowledge graph using pre-trained language model for learning entity-aware relationships. In *2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON)*, pages 310–315.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone. 2023. Llm+ p: Empowering large language models with optimal planning proficiency. *arXiv preprint arXiv:2304.11477*.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 36(7):3580–3599.

J Piaget. 1952. The origins of intelligence in children. *International University*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695.

Pratyusha Sharma, Antonio Torralba, and Jacob Andreas. 2022. Skill induction and planning with latent language. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1713–1726, Dublin, Ireland. Association for Computational Linguistics.

Tao Shen, Yi Mao, Pengcheng He, Guodong Long, Adam Trischler, and Weizhu Chen. 2020. Exploiting structured knowledge in text via graph-guided representation learning. *arXiv preprint arXiv:2004.14224*.

Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. 2023. Progprompt: Generating situated robot task plans using large language models. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11523–11530.

Yu Sun, Shuohuan Wang, Shikun Feng, Siyu Ding, Chao Pang, Junyuan Shang, Jiaxiang Liu, Xuyi Chen, Yanbin Zhao, Yuxiang Lu, et al. 2021. Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation. *arXiv preprint arXiv:2107.02137*.

Yi Tay, Mostafa Dehghani, Vinh Q Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Siamak Shakeri, Dara Bahri, Tal Schuster, et al. 2022. Ul2: Unifying language learning paradigms. *arXiv preprint arXiv:2205.05131*.

Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, Joao Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. 2023. Transformers learn in-context by gradient descent. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 35151–35174. PMLR.

Michihiro Yasunaga, Antoine Bosselut, Hongyu Ren, Xikun Zhang, Christopher D Manning, Percy S Liang, and Jure Leskovec. 2022. Deep bidirectional language-knowledge graph pretraining. In *Advances in Neural Information Processing Systems*, volume 35, pages 37309–37323. Curran Associates, Inc.

Donghan Yu, Chenguang Zhu, Yiming Yang, and Michael Zeng. 2022. Jaket: Joint pre-training of knowledge graph and language understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11630–11638.

Li Zhang, Hainiu Xu, Yue Yang, Shuyan Zhou, Weiqiu You, Manni Arora, and Chris Callison-Burch. 2023. Causal reasoning of entities and events in procedural texts. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 415–431, Dubrovnik, Croatia. Association for Computational Linguistics.