Intelligent Computing Technology and Automation
Z. Hou (Ed.)

doi:10.3233/ATDE231203

A Methodology for Generating and Optimizing Chain-of-Thought Based on Knowledge Graphs

Qiushi LUAN[a,1]

[a] *Shenyang Institute of Computing Technology, University of Chinese Academy of Sciences，Shen'yang,110168,China*

Abstract. One of the critical indicators for assessing the practical applicability of large language models is their competency in vertical domain question-answering tasks. However, in real-world applications, fine-tuning these large models often compromises their inherent capabilities. Moreover, fine-tuning does not offer precise control over the model's generated outputs.Consequently, enhancing the question-answering performance of large models in specialized domains has become a focal concern in the field. To address these challenges, this paper introduces a novel approach for generating and optimizing a "Chain-of-Thought"(CoT), leveraging domain-specific knowledge graphs. Specifically, we propose a Knowledge Graph-generated Chain of Thought (KGCoT) method that utilizes graph search algorithms to generate a chain of thought. This chain guides the injection of specialized knowledge into large language models and adapts the weightings based on user feedback, thereby optimizing subsequent graph searches.Heuristic searches are performed on the knowledge graph based on edge weights, culminating in the amalgamation of discovered entities and knowledge into a chain of thought. This KGCoT serves as a prompt to stimulate the large language model's contemplation of domain-specific knowledge. Additionally, an adaptive weight optimization formula refines the chain's weights in response to output feedback, thereby continually enhancing the quality of future search results and ensuring real-time optimization capabilities for the model.Through empirical evaluations conducted on publicly available datasets, the large language model ChatGLM, when prompted with a KGCoT, exhibited a 72.8% improvement in its BLEU score compared to its baseline performance. This outperformed other models like LLaMA and RWKV, unequivocally substantiating the efficacy of the proposed KGCoT method.

Keywords. Large Language Model; Knowledge Graph; Prompt; Chain-of-Thought; Optimize Search

1. Introduction

In recent years, advancements in deep learning and natural language processing (NLP) have propelled the development of large language models such as GPT and LaMDA, which have achieved remarkable successes across a variety of tasks. These models have instigated revolutionary changes in the NLP domain, substantially enhancing machine capabilities in understanding and generating human-like language.

[1] Corresponding Author:Qiushi Luan; Shenyang Institute of Computing Technology, University of Chinese Academy of Sciences, 110168, China; luantian1123@163.com

While these models excel in general tasks, their performance in specialized vertical sectors like healthcare and law remains suboptimal when untrained. The prevailing strategy to address this is through the fine-tuning of pre-trained models for these specific domains. However, despite the advancements in fine-tuning paradigms, the approach suffers from challenges like instability and catastrophic forgetting. Specifically, under identical hyperparameters, various initialization states can result in significant performance disparities for pre-trained language models. These discrepancies are particularly pronounced on smaller datasets where models exhibit poor generalization capabilities. Exacerbated by the limitations of the downstream dataset sizes, aggressive fine-tuning can trigger catastrophic forgetting and overfitting, leading to compromised performance when transitioning to out-of-domain data or related tasks [1-2].

Consequently, the integration of large models' parametric knowledge with structured knowledge from knowledge graphs has become an emergent area of focus. This synergy not only equips the models with supplementary information but also empowers them to leverage structured knowledge more efficiently for question-answering tasks, marking a promising direction in ongoing research [3].

Priyanka Sen and colleagues developed an end-to-end Knowledge Graph Question-Answering (KGQA) model based on the ReifKB and Rigel model families. This model is designed to return a weighted set of facts retrieved from a knowledge graph. Utilizing these retrieved facts as prompts in a zero-shot setting, the model then generates natural language responses. This methodology led to a significant enhancement in the model's average performance [4].

Jinheon Baek and associates proposed a method to amplify the capabilities of large language models by leveraging factual data from knowledge graphs. Operating on the principle of semantic similarity, this approach retrieves problem-related triples from the knowledge graph, transforms them into text strings, and then incorporates them as prompts into the language model's input. Remarkably, this technique resulted in a 48% improvement in model performance in zero-shot scenarios [5].

Takeshi Kojima and his team introduced a prompting technique named Chain-of-Thought (CoT). Rather than providing the model with conventional question-answer pairs, this technique stimulates complex multi-step reasoning by offering incremental reasoning examples. CoT prompts have been notably effective in tasks like arithmetic and symbolic reasoning, particularly when integrated with larger language models such as PaLM [6].

The aforementioned studies collectively validate the efficacy of augmenting large models with external knowledge sources and demonstrate the utility of Chain of Thought (CoT) prompts in boosting model performance. However, most existing works primarily focus on using triples from knowledge graphs as mere cues for subsequent large language models, without effectively incorporating the actual knowledge corresponding to the entities. Such an approach reveals only the relationships among pieces of knowledge but falls short of elucidating the intrinsic meaning of the knowledge or the rationale behind entity relationships. Furthermore, these studies do not fully assemble the pieces of knowledge into a coherent chain of thought for model prompting, thereby somewhat overlooking the integrity of logical reasoning from question to answer.

Moreover, for identical types of questions, there may be multiple potential search paths in the knowledge graph. However, due to variations in the prompting mechanisms, language models may exhibit considerable differences in their interpretive

abilities across these paths. More specifically, even when addressing the same precise question, the quality of the prompts derived from the graph can vary. Unfortunately, existing research has not optimized the graph itself to enhance its prompt effectiveness when integrated with natural language models.

Ruida Yang introduced a method for optimizing relationship weights in the knowledge graph based on user feedback. Initially, the Extended Inverse P-distance (EIPD) algorithm was employed to gauge the similarity between two entity nodes. Leveraging this algorithm, the problem of relationship weight optimization was reframed as a constraint optimization problem, leading to the formulation of a basic single-user feedback solution as well as an optimized multi-user feedback approach. Additionally, a "Segment-Aggregate" strategy was devised to boost computational efficiency. [7]

The aforementioned method fundamentally relies on entity similarity calculations and uses user feedback to provide candidate answers, essentially constituting a constraint optimization problem steered by the knowledge graph. However, when confronted with more open-ended questions, direct problem-solving becomes increasingly complex and sometimes infeasible.

As such, this study employs a heuristic search approach that uses probabilistic Depth-First Search (DFS) guided by edge weights and optimizes the search path probabilities based on user feedback. In contrast to a global optimization strategy, this method places more emphasis on the directional choices of nodes at each step of graph optimization. Overall, the method serves as a supplement to large language models for problem-solving by using the knowledge graph and inspires more accurate reasoning through a generated Knowledge Graph Chain of Thought (KGCoT).

In summary, this study introduces a method for Knowledge Graph-generated Chain of Thought (KGCoT) and proposes an adaptive weight optimization formula, which is driven by user feedback, to enhance the generated thought chain. Using the medical field as an example, the method extracts entities from the question, maps them onto a medical knowledge graph, and then performs heuristic DFS based on link weights. The discovered entities are indexed to their corresponding knowledge and assembled into a KGCoT to guide the large language model. Ultimately, this paper employs the method on the ChatGLM model and investigates the performance enhancement effects of the KGCoT on the model by evaluating the improved ChatGLM-KGCoT model on public datasets.

2. A Methodology for Generating and Optimizing Chain-of-Thought Based on Knowledge Graphs

2.1 Knowledge Graph Construction

This article employs two open-source datasets: the Knowledge Graph for Common Family Diseases [8] and Alibaba Tianchi's CHIP2022 Medical Causal Entity Relationship Dataset [9] to construct an extensive medical knowledge graph. The resulting graph includes a broad spectrum of medical entities, such as common diseases, symptoms, treatment modalities, frequently prescribed medications, and recommended dietary guidelines. It also encapsulates causal relationships between these entities, totaling 19,440 distinct medical entities and 47,105 inter-entity relationships.

In the creation of this knowledge graph, this article initially extracts entities from both datasets, utilizing Neo4j to establish the graph based on the associations among the entities found in the datasets. Concurrently, the corresponding knowledge for each entity is serialized into JSON format, and each is assigned a unique identifier to act as an article index. Considering that identical entities may have different relationships depending on the contextual knowledge, this index is stored as an edge attribute labeled "type" within the graph. Additionally, all edges are assigned an initial weight of 1 to simplify future graph traversals based on edge weights. The basic structure of the final constructed knowledge graph is shown in Figure 1.

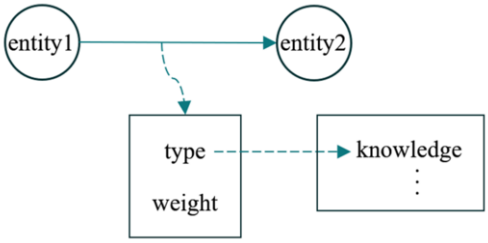

Figure 1. The Basic Structure of Knowledge Graph

2.2 Input data entity extraction

During the graph search phase, the first step is to extract pertinent medical entities and patient symptoms from the input sentences. To enhance the stability of entity extraction and boost the model's capability in this regard, this article runs a BertNER model in parallel with the foundational ChatGLM model.

This article employs Alibaba Tianchi's publicly available Chinese Medical Named Entity Recognition Dataset (CMeEE) [10] to train the BertNER entity extraction model. The dataset encompasses nine major categories of medical entities, including 504 common pediatric ailments, 7,085 anatomical locations, and 12,907 clinical manifestations.

To optimize the model, some of the weights in the Bert-NER model are frozen and the model is fine-tuned through 100 training epochs focused on entity extraction. As a result, the trained BertNER model gains the proficiency to extract entities within the medical domain.

Additionally, leveraging the generalization capability of large models for downstream tasks, this article introduces an extraction template designed for gathering patient physiological indicators. Utilizing a Few-shot learning approach, the model is supplied with several case examples to train on. Consequently, the model learns to identify and return physiological elements from the input data in a manner consistent with the examples provided. The format for these extraction cases is depicted in Figure2.

Entities = '性别，年龄，身高，体重，病程，数值指标，已做检查，已用药物，个人疾病历史，家族疾病历史，个人情况'
(Entities = 'gender, age, height, weight, disease duration, numerical indicators, checked, medications used, personal disease history, family disease history, personal situation')

问句示例： f '请从以下语句中："""医生，我今年25岁，身高160, 体重130，前几天刚查出了糖尿病，空腹血糖5.2，糖化5.2，餐后2h却达到了11.3，医生没有给我开药，只是叫我先控制饮食，减肥，我有点害怕并发症，我今天喝了3杯400ml左右的水，和一瓶500ml左右的无糖饮料，但是只去了3次厕所，每次尿量特别少"""提取出以下信息："""{Entities}"""'
(Examples of questions : f 'Please refer to the following sentence:" "Doctor, I am 25 years old, 160 in height and 130 in weight. I just found out diabetes a few days ago. The fasting blood sugar was 5.2, and the saccharification was 5.2. But it reached 11.3 hours after the meal. The doctor didn't prescribe medicine for me, but asked me to control my diet first and lose weight. I am a little afraid of complications. Today, I drank three glasses of 400ml water and a bottle of 500ml sugar free drink, but I only went to the toilet three times, and the urine volume was very small each time" " Extract the following information: """ {Entities}""" '

回答示例： '年龄：25岁；性别：无；身高：160；体重：130；病程：前几天；数值指标：空腹血糖5.2、糖化5.2、餐后2h11.3；已做检查：血糖；已用药物：无；个人疾病历史：糖尿病；家族疾病历史：无；个人情况：尿量少'
(Examples of responses : 'Age: 25 years old; Gender: None; Height: 160; Weight: 130; Course of disease: a few days ago; Numerical indicators: fasting blood glucose 5.2, glycosylated hemoglobin 5.2, postprandial 2h11.3; Inspection has been conducted: blood sugar; Drugs used: none; Personal disease history: diabetes; Family disease history: none; Personal situation: low urine output')

Figure 2. Few-shot method extraction example

Upon completing the extraction of input data, all entities identified by Bert-NER are combined with the information obtained through ChatGLM, excluding numerical indicators such as gender, age, and height. After character filtering and deduplication processes, the remaining entities are aligned within the knowledge graph for the purpose of generating a chain of thought. The overall structure of the entity extraction part is shown in Figure 3.

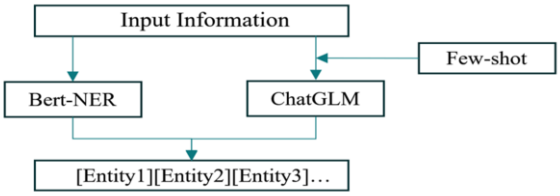

Figure 3. Overall structure of entity extraction part

2.3 KGCoT Generation and Integration

To facilitate the alignment of extracted entities with the knowledge graph, this study employs the open-source datasets previously used for knowledge graph construction: a knowledge graph focused on common family diseases and Alibaba Tianchi's CHIP2022 Medical Causal Entity Relationship Dataset. These datasets serve as the basis for training an entity alignment Word2Vec model. To enable directional mapping of entities extracted from input sentences to those already existing in the graph, it is essential to store interrelated entities in the datasets as adjacent words. These are then subjected to Word2Vec training via the Skip-gram method. After completing 50 training epochs, the Word2Vec model becomes capable of mapping extracted entities to their corresponding entities within the graph. Entities slated for querying are sequentially input into the trained Word2Vec model, aligning them with relevant entities in the graph. These entities then serve as initial nodes for executing a depth-first search through the graph, based on edge weights.

Assuming that starting from the current node, there are l available paths represented as $l=\{l_1,l_2,l_3,...\}$, with corresponding weights $\omega=\{\omega_1,\omega_2,\omega_3,...\}$. To prevent redundancy in the generated chain of thought, it's

necessary to record the identifiers of the paths already traversed. Let $N=\{n_1,n_2,n_3,...\}$ represent the set of such recorded paths. Consequently, for the current node, the probability p_i of selecting the adjacent i-th path is determined.

$$p_i=\begin{cases}\dfrac{\omega_i}{\sum_{j=1}^{n}\omega_j} & ,l_i\notin N\\ 0 & ,l_i\in N\end{cases} \tag{1}$$

Upon selecting a subsequent path, it's imperative to document the next encountered entity, denoted as e_i. The pertinent knowledge is then retrieved using the article index embedded within the graph edge. By concatenating all such discovered entities into a KGCoT $e_1\rightarrow e_2\rightarrow e_3$,we concurrently obtain the associated body of knowledge, represented as $k_1\rightarrow k_2\rightarrow k_3$ The process of KGCoT generation and integration is shown in Figure 4.

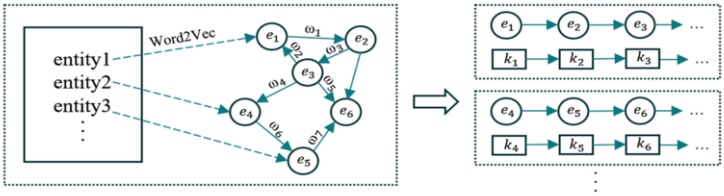

Figure 4. Example of KGCoT integration

To facilitate future refinements to the knowledge graph, it's essential to log the link ID corresponding to each KGCoT when selecting an entity. This enables the model to dynamically adjust the weights of individual links based on subsequent feedback.

Upon obtaining the KGCoT and its corresponding body of knowledge, both are stored in the model's history as question-and-answer pairs. By utilizing customized prompt templates, the model is guided to comprehend the sequence and salience of the knowledge, aligned with the chain's structure, thereby effectively responding to user queries. Figure 5 showcases an illustrative example designed to prompt the model for knowledge interpretation based on the KGCoT.

Prompt1 = (‘请问该如何使用思维链？’, ‘思维链是按照逻辑顺序串联起来的实体，通过'->'来标识思维链的方向’)
(**Prompt1** = 'How can I use the chain-of-thought?', 'Chain-of-Thought is an entity connected in a logical order. '->' is used to identify the direction of chain-of-thought. ‘)

Prompt2 = f(‘请问第{i}条思维链"""{chain}"""对应的知识都有哪些’, f’ 第{i}条思维链对应的知识为：{knowledge}’)
(**Prompt1** = f' What are the knowledge corresponding to the {i} chain-of-thought""" {chain} """? The knowledge corresponding to the {i} thinking chain is: {knowledge})

Prompt3 = f'请参照思维链理解其对应的知识，回答以下问题："""{input_text}"""'
(**Prompt3** = f 'Please refer to the thinking chain to understand its corresponding knowledge and answer the following question: """ {input_text} """)

Figure 5. Examples of KGCoT guidance model

2.4 Feedback-based KGCoT optimization

Upon integrating knowledge from the graph to produce responses, the model's output is subjected to human quality assessment. If the answer receives an "excellent" rating during this evaluation, the system leverages the stored ID of the KGCoT to identify its corresponding edge in the knowledge graph, subsequently boosting the weight of that KGCoT. Conversely, if the evaluation deems the answer to be "unsatisfactory," the weight assigned to the respective KGCoT is diminished.

For a given node e_i , assume it has more than two weighted adjacent nodes (if not, no modification is required). Let l_i be the edge for which the weight needs to be increased, with its corresponding weight denoted as ω_i, let $S = \sum_{j=1}^{n} \omega_j$ represent the sum of the weights of all adjacent edges. The probability of selecting this node prior to feedback adjustment is as follows:

$$p_i = \frac{\omega_i}{\sum_{j=1}^{n} \omega_j} = \frac{\omega_i}{S} \tag{2}$$

When the feedback is rated as "excellent," the weight increase for the edge is designated as $\omega_{increase}$, such that:

$$\omega_{increase} = \frac{\alpha(1 - p_i)}{1 - \alpha(1 - p_i)} \cdot S \tag{3}$$

The boundary factor $\alpha \in (0, 1)$ serves as the upper limit for the increased likelihood of an edge being selected as its weight augments, specifically when the probability p_i of selecting the node's edge tends towards zero. Additionally, when p_i approaches 1, $\omega_{increase}$ asymptotically approaches zero, thereby preventing infinite weight accumulation for a single node.In practical applications, to avert the issue of competitive weight accumulation across multiple nodes — a situation where the selection probability p remains constant, yet the sum of the weights S increases — the boundary factor α is commonly set to $\frac{1}{S}$. This effectively caps the upper limit for weight fluctuations. Lastly, concerning the edge l_i , after the weight $\omega_{increase}$ is added, the new probability of this edge being selected shifts to p_i^* .

$$p_i^* = \frac{\omega_i + \omega_{increase}}{S + \omega_{increase}} \tag{4}$$

Assuming the change in the probability of selection after altering its weight is denoted by Δp_i ,then:

$$|\Delta p_i| = p_i^* - p_i = \alpha(1 - p_i)^2 \tag{5}$$

For edge l_i , increasing its weight by $\omega_{increase}$ will result in different behaviors depending on its initial selection probability p_i .Specifically, as p_i approaches zero, the change in the selection probability approaches the boundary factor α . On the other hand, as p_i approaches one, the change in the selection probability $|\Delta p_i|$ tends toward zero. Concurrently, as p_i increases, the rate of change in the probability difference $|\Delta p_i|'$ tends toward zero. When the feedback result is deemed "poor", the weight reduction for the edge is set as $\omega_{decrease}$,where:

$$\omega_{decrease} = \frac{\beta p_i^2}{1 - p_i + \beta p_i^2} \tag{6}$$

In this context, the boundary factor $\beta \in (0, 1)$ signifies the maximum decline in the probability of edge selection as its weight decreases when the initial selection probability p_i nears 1. Importantly, $\omega_{decrease}$ is always less than the original weight value ω_i , ensuring that the adjusted weight remains positive. Moreover, after decreasing the weight $\omega_{decrease}$ of edge l_i ,the new probability of this edge being selected shifts to p_i^* .

$$p_i^* = \frac{\omega_i - \omega_{decrease}}{S - \omega_{decrease}} \tag{7}$$

Let Δp_i be the change in the probability of selection after adjusting the weight; in this case:

$$|\Delta p_i| = p_i - p_i^* = \beta p_i^2 \tag{8}$$

For edge l_i, if its weight is reduced by $\omega_{decrease}$, then as its initial selection probability p_i tends toward 0, the change $|\Delta p_i|$ in its selection probability also tends toward 0. Concurrently, as p_i decreases, the rate of change of the probability difference, denoted as $|\Delta p_i|'$ progressively diminishes. When the initial selection probability p_i is close to 1, the variation in the selection

probability approximates β. To sum up, the structure of the KGCoT generation and optimization algorithm based on knowledge map is shown in Figure 6.

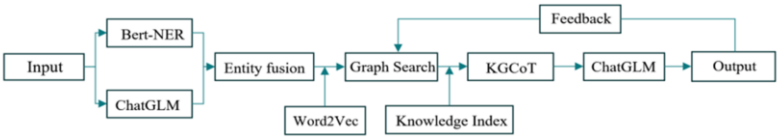

Figure 6. KGCoT generation and optimization structure

3. Experimental Study On

3.1 Preparation

Our experimental setup utilizes Ubuntu 11.3.0. For GPU, we equipped the system with one NVIDIA GeForce RTX 3090 (24GB) and three Tesla M40 24GB units. The CPU in use is an Intel(R) Xeon(R) E5-2686 v4, clocking in at 2.30GHz. We employed CUDA version 11.7 and used public medical QA datasets, specifically Chinese-medical-dialogue-dat[11] and cMedQA2[12]. Our test domains encompassed andrology, internal medicine, obstetrics and gynecology, oncology, pediatrics, and surgery.

3.2 Experimental Scheme

This study employs BLEU, a classic metric for assessing machine translation quality, and the ROUGE metric, frequently used in machine translation, automatic summarization, and question-answer generation, as evaluation standards. For comparative experiments, we juxtapose the ChatGLM-6B model enhanced with chain-of-thought prompts and weight updates (ChatGLM-KGCoT) against several others: the original ChatGLM-6B, the ChatGLM-6B model fine-tuned with LoRA, the ChatGLM-6B model fine-tuned with P-tuning, the ChatGLM2-6B model, RWKV-7B model, and the LLaMa-7B-Chinese model, comparing their performance across two datasets. To verify the enhancing effect of the feedback mechanism on KGCoT, the ChatGLM-KGCoT model assesses the current BLEU score in real-time after each inference, subsequently optimizing the KGCoT. To assess the model's performance on the Chinese-medical-dialogue-data dataset, we randomly select 100 entries from each department classification, totaling 600 entries for testing. For the cMedQA2 dataset, we randomly choose 100 entries for evaluation. For the ChatGLM model's LoRA and P-tuning adjustments, we use an 80-20 split, where 80% of the data is for training and 20% for testing. The scores of ChatGLM, ChatGLM-LoRA, and ChatGLM-Ptuning models on the Chinese-medical-dialogue-data dataset are cited from publicly available results for this dataset. For ablation studies, we contrast the ChatGLM-6B model, incorporating KGCoT prompts and weight updates, with ChatGLM-1, which seeks knowledge purely based on weight and conveys it to the ChatGLM model without KGCoT prompts, and ChatGLM-2, which dispenses with weight updates and randomly picks nodes to convey knowledge to the ChatGLM model through KGCoT. We also

include the original ChatGLM model for comparison, aiming to discern the effects of KGCoT guidance and weight updates within the model.

3.3 Experimental Results

The score performance of the model on the two test sets is shown in Table 1 :

Table 1. Comparative test of model generation index

Model name	Chinese-medical-dialogue-data				cMedQA2			
	BLEU-4	Rouge-1	Rouge-2	Rouge-L	BLEU-4	Rouge-1	Rouge-2	Rouge-L
ChatGLM	3.21	17.19	3.07	15.47	2.42	26.40	3.95	22.31
ChatGLM-LoRA	4.21	17.74	3.56	16.61	3.49	18.07	2.47	16.50
ChatGLM-Ptuning	3.55	18.24	2.74	15.02	5.28	19.47	3.81	17.44
ChatGLM2	3.53	25.34	3.22	19.01	2.25	26.74	3.92	22.44
RWKV	3.54	16.50	2.04	13.78	3.26	19.64	2.11	18.16
LLaMa2-Chinese	1.22	20.07	1.27	16.59	0.64	21.61	1.12	18.57
ChatGLM-KGCoT	5.31	25.85	5.01	22.85	4.42	30.29	6.47	27.03

Comparing the BLEU scores of the ChatGLM-KGCoT model with the native ChatGLM model as shown in Table 1, it's evident that after implementing KGCoT prompts and optimizations, the ChatGLM model's BLEU score improved by 65.4% on the Chinese-medical-dialogue-data dataset and by a staggering 82.6% on the cMedQA2 dataset. On the whole, when juxtaposed with the original model, the enhanced model witnessed an average BLEU score surge of 72.8%. This underscores that the post-prompt ChatGLM model enhanced both its output quality and lexical precision. Further juxtaposing the ChatGLM-KGCoT with other models, it's observable that the optimized ChatGLM model's BLEU score rose by an average of 26.4% when compared with the lora-fine-tuned model and by 10.2% against the p-tuning fine-tuned model. This suggests that integrating external knowledge into models via KGCoT prompts is potentially superior or at least on par with embedding parameterized knowledge through micro-tuning.

A side-by-side comparison with ChatGLM2 and RWKV reveals that ChatGLM-KGCoT holds an average leading edge in BLEU scores by percentages of 68.3% and 43.1% on the two datasets, and far exceeds the LLaMa2-Chinese model. The prominent performance of the KGCoT model, especially when compared to RWKV and ChatGLM2 models, endorses the efficacy of the KGCoT method. The LLaMa2 model's relatively lower score might be attributed to its roots as an English-centric model, implying that its performance in Chinese, even after fine-tuning, doesn't quite meet expectations. Regarding the Rouge metric, the ChatGLM-KGCoT model, when compared to its predecessor, showcased average increments of 28.8%, 63.5%, and 32% across Rouge-1, Rouge-2, and Rouge-L respectively. This buttresses the claim that the KGCoT prompted model possesses a more refined knack for pinpointing and controlling key terms in both the questions and their corresponding answers, making the generated responses more aligned with the original queries. Moreover, when set against other models, the ChatGLM-KGCoT demonstrates a significant lead in scores. In conclusion, the ChatGLM-KGCoT model, in comparison to its counterparts, unequivocally displays a competitive edge in the Q&A domain. This can be credited to the KGCoT prompts, which, to some extent, bolster the model's grip on Q&A key terms. Post integration of the knowledge graph, the vocabulary in the model's responses that correlates with the key terms in the questions has also seen an

uptick, a phenomenon corroborated by results across both datasets. The results of the ablation experiment are shown in Table 2.

Table 2. Ablation Experiment of model generation index.

Model name	Chinese-medical-dialogue-data				cMedQA2			
	BLEU-4	Rouge-1	Rouge-2	Rouge-L	BLEU-4	Rouge-1	Rouge-2	Rouge-L
ChatGLM	3.21	17.19	3.07	15.47	2.42	26.40	3.95	22.31
ChatGLM-1	4.44	24.88	4.15	22.41	3.08	27.20	4.53	24.51
ChatGLM-2	4.33	23.76	3.77	21.14	2.70	28.20	4.57	26.17
ChatGLM-KGCoT	5.31	25.85	5.01	22.85	4.42	30.29	6.47	27.03

Based on the results from the ablation experiments, after removing the KGCoT prompts but retaining the weighted search mechanism, the BLEU score of the ChatGLM-1 model decreased by an average of 22.7% across the two datasets when compared to the ChatGLM-KGCoT model. Additionally, the Rouge-1, Rouge-2, and Rouge-L scores decreased by averages of 7.2%, 24.4%, and 5.9% respectively. However, when juxtaposed against the original ChatGLM model, even without the KGCoT but with weight-based search, the ChatGLM-1 model demonstrated a 33.6% surge in its BLEU score and improvements of 19.5%, 23.6%, and 24.2% in Rouge-1, Rouge-2, and Rouge-L metrics respectively. The decline in scores for the ChatGLM-1 model after removing the KGCoT suggests that employing a KGCoT, generated via the knowledge graph, as a guidance mechanism for the model is effective. The score enhancements of ChatGLM-1, which used weight-based search when compared to the baseline ChatGLM, validate that heuristic knowledge search based on weights can meaningfully elevate model performance. On the other hand, the ChatGLM-2 model, which eliminated edge weights but randomly picked adjacent nodes while still maintaining the KGCoT prompts, trailed the ChatGLM-KGCoT model with a 27.7% drop in its average BLEU scores across both datasets. Rouge-1, Rouge-2, and Rouge-L scores also plummeted by 7.4%, 27.4%, and 5.2% respectively. These results underscore that, compared to a random knowledge graph search, optimizing the KGCoT weights with feedback can substantively enhance the quality of the model's responses. Against the baseline ChatGLM model, the ChatGLM-2 model experienced an average boost of 24.9% in its BLEU score and increments of 19.2%, 18.8%, and 25.2% in Rouge-1, Rouge-2, and Rouge-L metrics respectively. This signifies that prompting the model with entities consolidated into a KGCoT and guiding the model's thought processes can beneficially modulate the model's answering capabilities. In conclusion, the performance disparity between ChatGLM-1, which removed KGCoT but maintained weight-based search, and ChatGLM-KGCoT, along with the comparison between ChatGLM-2, which held onto the KGCoT but discarded weight-based search, and the baseline ChatGLM, collectively attest to the effectiveness of guiding model cognition through KGCoT generated by the knowledge graph. Moreover, the enhancements observed in the ChatGLM-1 model relative to the native ChatGLM and the deficits noted in the ChatGLM-2 model when contrasted with ChatGLM-KGCoT reaffirm that heuristic knowledge searches based on weights can tangibly uplift model performance.

4. Conclusion

In recent years, as natural language processing technology has advanced, the focus of research has increasingly shifted toward pre-trained large language models. Enhancing

the capabilities of these large models in specialized domains has become a central concern. Distinct from fine-tuning approaches, integrating knowledge graphs with large models serves the dual purpose of infusing models with domain-specific knowledge while mitigating model 'hallucinations.' This paper introduces a Knowledge Graph-based Chain of Thought (KGCoT) generation method. Leveraging probabilistic depth-first search algorithms weighted by graph edges, this method produces KGCoT aligned with specific questions. These KGCoT, along with their associated knowledge, are subsequently fed into the model via prompts, enabling heuristic modes of problem-solving rooted in the knowledge graph. Furthermore, the paper proposes a self-adaptive graph weight optimization algorithm that provides evaluative feedback concurrent with model output. This feedback-driven graph dynamically adjusts edge weights to increase or decrease the likelihood of reactivating particular KGCoT. Overall, this research offers an efficacious methodology for applying large language models to specialized domain question-answering tasks. By melding knowledge graphs with KGCoT and facilitating their optimization, the method has empirically demonstrated improvements in the vertical domain question-answering performance of large models. This approach opens new avenues for integrating large models with knowledge graphs in specialized domains, while also identifying future research opportunities for improving the model-graph synergy at a structural level.

References

[1] Cheolhyoung Lee, Kyunghyun Cho, and Wanmo Kang. Mixout: Effective regularization to finetune large-scale pretrained language models. In ICLR, 2020,1-17

[2] Xu R , Luo F , Zhang Z ,et al.Raise a Child in Large Language Model: Towards Effective and Generalizable Fine-tuning. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 2021,9514-9528

[3] Pan, Shirui et al. “Unifying Large Language Models and Knowledge Graphs: A Roadmap.” ArXiv abs/2306.08302 (2023): n. pag.

[4] Sen P, Mavadia S, Saffari A. Knowledge graph-augmented language models for complex question answering.1st Workshop on Natural Language Reasoning and Structured Explanations (NLRSE), 2023,1-8

[5] Baek J, Aji A F, Saffari A. Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering[C].1st Workshop on Natural Language Reasoning and Structured Explanations (NLRSE),2023,pp.78-106

[6] Kojima T, Gu S S, Reid M, et al. Large language models are zero-shot reasoners.Advances in neural information processing systems, 2022, 35: 22199-22213.

[7] Yang Ruida. Research and application of knowledge graph relationship optimization technology based on user feedback information[D]. East China Normal University,2019,09:91(in Chinese)

[8] Wang Zhichang, Luo Zhuoyan, Zhu Qipeng, Zhu Haojia, Wang Xiangyuan, Wu Tianxing. Knowledge map for common family diseases.[online] http://openkg.cn/dataset/medicalgraph (in Chinese)

[9] Li Zihao, Chen Mosha, Ma Zhenxin, et al. Chinese medical causality extraction data set CMedCausal. Journal of Medical Informatics,2022,43(12):23-27.(in Chinese)

[10] Hongying Zan, Wenxin Li, Kunli Zhang, Yajuan Ye, Baobao Chang, Zhifang Sui. "Building a Pediatric Medical Corpus: Word Segmentation and Named Entity Annotation." In Workshop on Chinese Lexical Semantics, 2020,652-664,

[11] Toyhom. Chinese-medical-dialogue-data. 2019.[online] https://github.com/Toyhom/Chinese-medical-dialogue-data

[12] Zhang, S.; Zhang, X.; Wang, H.; Cheng, J.; Li, P.; Ding, Z. Chinese Medical Question Answer Matching Using End-to-End Character-Level Multi-Scale CNNs. Appl. Sci. 2017, 7, 767,[online] https://doi.org/10.3390/app7080767