KaLM: Knowledge-aligned Autoregressive Language Modeling via Dual-view Knowledge Graph Contrastive Learning

Anonymous ACL submission

Abstract

Autoregressive large language models (LLMs) pre-trained by next token prediction are inherently proficient in generative tasks. However, their performance on knowledge-driven tasks such as factual knowledge querying remains unsatisfactory. Knowledge graphs (KGs), as high-quality structured knowledge bases, can provide reliable knowledge for LLMs, potentially compensating for their knowledge deficiencies. Aligning LLMs with explicit, structured knowledge from KGs has been a challenge; previous attempts either failed to effectively align knowledge representations or compromised the generative capabilities of LLMs, leading to less-than-optimal outcomes. This paper proposes **KaLM**, a *Knowledge-aligned Language Modeling* approach, which fine-tunes autoregressive LLMs to align with KG knowledge via the joint objective of explicit knowledge alignment and implicit knowledge alignment. The explicit knowledge alignment objective aims to directly optimize the knowledge representation of LLMs through dual-view knowledge graph contrastive learning. The implicit knowledge alignment objective focuses on incorporating textual patterns of knowledge into LLMs through triple completion language modeling. Notably, our method achieves a significant performance boost in evaluations of knowledge-driven tasks, specifically embedding-based knowledge graph completion and generation-based knowledge graph question answering[1].

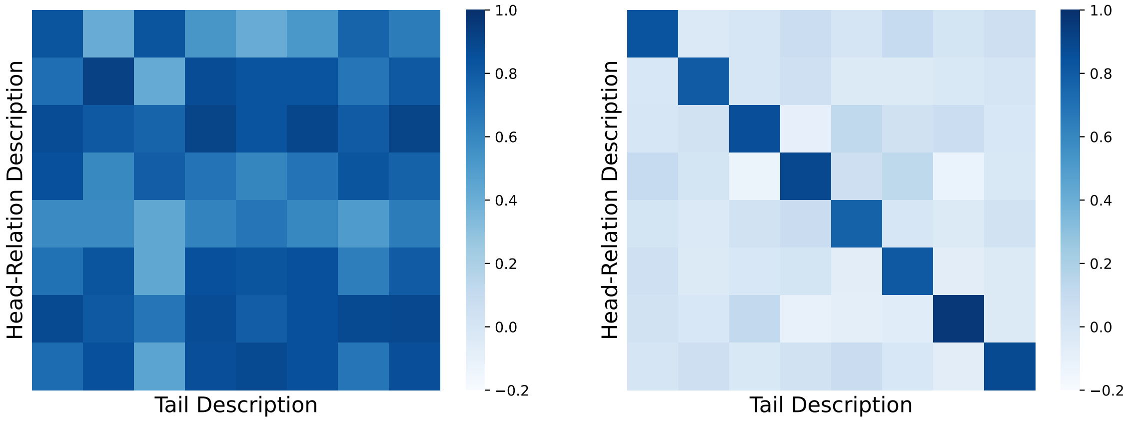
1 Introduction

Large language models (LLMs) like PaLM 2 (Anil et al., 2023) and GPT-4 (Achiam et al., 2023) have recently made remarkable advancements in a wide range of natural language processing tasks (Li et al., 2022; Su et al., 2019). However, LLMs still face challenges in tasks requiring factual or domain-specific knowledge, resulting in unsatisfactory performance in knowledge-driven tasks. From the perspective of knowledge representation, LLMs serve as parametric knowledge bases, providing implicit, non-deterministic knowledge, while knowledge graphs (KGs) function as structured knowledge bases, offering explicit, deterministic knowledge. KGs, commonly organized as factual knowledge triples describing relations between entities, can serve as a reliable knowledge source for LLMs. Aligning LLMs with KG knowledge can enhance the knowledge reasoning capabilities of LLMs and improve their performance on knowledge-driven tasks, such as knowledge graph completion (KGC) and knowledge graph question answering (KGQA).

Autoregressive LLMs pre-trained through next token prediction tasks often exhibit limitations in knowledge representation, leading to embeddings that lack diversity and specificity. This limitation becomes evident in tasks that demand distinctive sentence embeddings, such as dense retrieval and semantic search (Muennighoff, 2022; Ma et al., 2023). As demonstrated in Figure 1(a), the representations generated by LLMs tend to be overly homogeneous across different pieces of knowledge, undermining their effectiveness in applications requiring fine-grained semantic distinctions.

The concept of explicit knowledge alignment is introduced to directly optimize the knowledge representation within language models by devising direct knowledge training objectives. This strategy emerges in response to the observed degradation in knowledge representation within autoencoder-based pre-trained language models (PLMs), a phenomenon termed *representation anisotropy* (Ethayarajh, 2019). This issue is characterized by the clustering of learned token and sentence embeddings within a constrained area of the representation space, leading to a lack of distributional uniformity (Li et al., 2020). While previous efforts to address representation anisotropy have largely concentrated on promoting uniformity among to-

[1]Our code is available at https://anonymous.4open.science/r/KaLM-ARR.



(a) LLaMA

(b) KaLM

Figure 1: Similarity matrix of knowledge representations of (a) LLaMA and (b) KaLM. The values denote the cosine similarity between the head-relation embedding and tail embedding. The diagonal elements represent positive <head-relation, tail> pairs from the same KG triple, which should maintain high similarity (darker color); off-diagonal elements represent negative <head-relation, tail> pairs from different KG triples, which should have lower similarity (lighter color). In an ideal setting, knowledge representations should be able to distinguish between different triples, while maintaining alignment and uniformity of the representation, as shown in Figure 1(b).

ken representations, they often overlook the critical alignment of similar sentence representations (Su et al., 2021; Li et al., 2020; Su et al., 2022). More recent works advocate for integrating KG triples and using knowledge graph embedding losses to fine-tune PLMs, aiming to bolster their knowledge representation abilities (Shen et al., 2022; Wang et al., 2022b). Nonetheless, such approaches may limit themselves to optimizing at the token level or reduce the model to a mere text encoder, thereby diminishing its inherent generative capabilities.

Conversely, implicit knowledge alignment leverages the pre-training or fine-tuning of language models with external knowledge sources, employing the vanilla language modeling objective or its variations. This approach predominantly preserves the next token prediction framework, essentially retaining the native text generation prowess of LLMs. In the realm of implicit knowledge alignment, the prevalent practice involves the fine-tuning of LLMs with KG triples and their textual descriptions, as opposed to directly altering the hidden knowledge representations (Chen et al., 2022; Yao et al., 2023). Nevertheless, the efficacy of these methods on knowledge graph completion tasks remains substantially inferior when compared to strategies that directly fine-tune knowledge representations (Wang et al., 2022b,a). Intriguing findings from (Fu et al., 2023) reveal that fine-tuning PLMs with randomly unaligned KG triples can achieve performance on par with that obtained through fine-tuning with aligned triples in various tasks, including named entity recognition and relation classification. Their findings suggest that the hidden states of entities, whether infused with aligned or random knowledge, exhibit remarkable similarity. Consequently, existing implicit alignment methods fail to effectively utilize the injected knowledge or accurately discern the connection between newly introduced knowledge and the model's inherent knowledge, culminating in suboptimal performance.

In this paper, we propose **KaLM**, a *Knowledge-aligned Language Modeling* approach for aligning LLMs with KG knowledge. Specifically, we use KG triples and their textual descriptions to fine-tune LLMs via the joint objective of *explicit knowledge alignment* and *implicit knowledge alignment.*

The explicit knowledge alignment objective aims to directly optimize the hidden representations of knowledge in LLMs through *dual-view knowledge graph contrastive learning*. We theoretically prove and empirically show that this objective can facilitate knowledge representation alignment and alleviate representation anisotropy. For KG triples, we consider tail entity description and the concatenation of head entity description and relation description as two distinct views of the same knowledge. *The key insight is that: (1) representations of two different views of the same knowledge (i.e., from the same triple) should be pulled together, while (2)* *representations of different knowledge (i.e., from different triples) should be pushed apart.* The first term encourages semantically similar knowledge to remain close in the representation space, promoting knowledge representation alignment. The second term forces dissimilar knowledge to be as far apart as possible in the vector space, improving knowledge representation uniformity and mitigating representation anisotropy. As shown in Figure 1(b), our method can obtain the ideal knowledge representations that are both aligned and uniform.

The implicit knowledge alignment objective focuses on incorporating textual patterns of knowledge into LLMs through *triple completion language modeling*, which can maintain the generative capability of LLMs and boost performance on knowledge inference tasks. We constructed a triple completion dataset based on the KG triples to fine-tune LLMs, improving their instruction-following ability and facilitating implicit knowledge alignment. We also show the implicit knowledge alignment objective can further boost knowledge representation performance. This confirms that both explicit alignment and implicit alignment are crucial for knowledge alignment, as they both essentially require a deep understanding of knowledge.

Our contributions are summarized as follows:

- We introduce **KaLM**, a *knowledge-aligned language modeling* approach that aligns autoregressive LLMs with KG knowledge via the joint objective of *explicit knowledge alignment* and *implicit knowledge alignment*.
- We *theoretically prove and empirically demonstrate* that the explicit knowledge alignment objective achieved through dual-view knowledge graph contrastive learning can facilitate knowledge representation alignment and alleviate the issue of representation anisotropy.
- The experimental results on knowledge-driven tasks demonstrate the effectiveness of *KaLM*. In the embedding-based KGC task, KaLM significantly improves Mean Rank and Hit@10 metrics compared to previous state-of-the-art methods. In the generation-based KGQA task, KaLM achieves a notable improvement in answering accuracy compared to the base LLM.

2 Related Work

Our work is closely related to Knowledge Enhancement for LLMs and Representation Anisotropy of Language Models. A more detailed review of related work can be found in Appendix A.

Knowledge Enhancement for LLMs Knowledge enhancement aims to incorporate factual and domain-specific knowledge into LLMs to address their knowledge deficiencies. This can be divided into retrieval-based augmentation and training-based integration. *Retrieval-based knowledge augmentation* methods leverage external retrieval modules to provide additional knowledge, aiming to improve the knowledge reasoning capability of LLMs (Sun et al., 2023; Jiang et al., 2023). However, this approach may lead to knowledge conflicts (Feng et al., 2023), where knowledge in LLMs and knowledge in the retrieved documents are inconsistent or the retrieved multiple documents are contradictory. *Training-based knowledge integration* methods involve using KG triple descriptions to pre-train or fine-tune LLMs, aiming to achieve knowledge alignment. These methods can be divided into explicit alignment (Wang et al., 2021b; Yasunaga et al., 2022) and implicit alignment (Yao et al., 2023; Zhang et al., 2023) based on whether they directly optimize the knowledge representation. Nevertheless, prior methods have either sacrificed the generative capability or lacked effective representation alignment. Our approach enhances the knowledge of LLMs via a unique joint objective of explicit alignment and implicit alignment, improving the quality of knowledge representations and generative knowledge reasoning capabilities.

Representation Anisotropy of Language Models PLMs have long been plagued by representation anisotropy (Ethayarajh, 2019), where the learned token and sentence embeddings are confined to a narrow cone within the entire representation space. The issue of representation anisotropy not only results in model degradation (Su et al., 2022) but also leads to poor performance on discriminative tasks. Previous work on alleviating representation anisotropy has mainly focused on post-processing techniques such as normalizing flows (Li et al., 2020) or whitening operations (Su et al., 2021). Su et al. (2022) propose a contrastive training objective to encourage learning isotropic token representations. However, these methods mainly improve the isotropy of token representations without enhancing the discriminability of sentence representations. Our method improves the token-level and sentence-level representation anisotropy of LLMs through dual-view knowledge graph contrastive learning, and it has rigorous theoretical guarantees.

3 Knowledge-aligned Autoregressive Language Modeling

In this section, we introduce **KaLM**, a *Knowledge-aligned Language Modeling* approach for aligning LLMs with KG knowledge via the joint objective of *explicit knowledge alignment* and *implicit knowledge alignment*. The overview is shown in Figure 2.

3.1 Notations and Preliminaries

A KG $\mathcal{G}$ stores factual knowledge, denoted as $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T}, \mathcal{D})$. $\mathcal{E}$ and $\mathcal{R}$ are the set of entities and relations, respectively. $\mathcal{D}$ is the description set of all entities and relations. $\mathcal{D}_e$ and $\mathcal{D}_r$ are the textual description of entity e and relation r, respectively. $\mathcal{T} = \{(h, r, t)|h, t \in \mathcal{E}, r \in \mathcal{R}\}$ is the triple set. A triple (h, r, t) depicts the fact that there is a relation r between the head entity h and the tail entity t.

3.2 Explicit Knowledge Alignment

For KG triples, the textual description of the tail entity and the concatenation of the textual descriptions of the head entity and relation can be seen as two distinct views of the same knowledge. This inspires *KaLM* to align representations of two distinct views of the same knowledge (i.e., from the same triple), while separating representations of different knowledge (i.e., from different triples).

The LLM, denoted as E_{LLM}, is fine-tuned with the *dual-view knowledge graph contrastive learning* loss. The training corpus contains paired textual descriptions, $\{(\mathcal{D}_{hr}, \mathcal{D}_t)\}_{i=1}^{N}$, where $\mathcal{D}_t$ is the tail entity description, and $\mathcal{D}_{hr}$ is the concatenation of the head entity description and relation description. Given a training pair $(\mathcal{D}_{hr}, \mathcal{D}_t)$, the same E_{LLM} is used to compute the embeddings of $\mathcal{D}_{hr}$ and $\mathcal{D}_t$ independently. Moreover, we prepend the `[bos]` token to the beginning and append the `[eos]` token to the end of the textual description. The augmented input is fed into E_{LLM}, and the hidden representation corresponding to the `[eos]` token from the last layer is used as the final embedding of the input.

$$e_{hr} = E_{LLM}(\texttt{[bos]}_{hr} \oplus \mathcal{D}_{hr} \oplus \texttt{[eos]}_{hr}),$$
$$e_t = E_{LLM}(\texttt{[bos]}_t \oplus \mathcal{D}_t \oplus \texttt{[eos]}_t),$$

where $\oplus$ is the operation to concatenate two strings and $\mathcal{D}_{hr} = \mathcal{D}_h \oplus \mathcal{D}_r$. For stable training, we adopt "[" as $\texttt{[bos]}_{hr}$ and "]" as $\texttt{[eos]}_{hr}$, while using "{" as $\texttt{[bos]}_t$ and "}" as $\texttt{[eos]}_t$.

We utilize the knowledge graph contrastive learning loss to directly optimize the knowledge representation of the LLM by *encouraging semantically similar knowledge to stay close in the representation space and pushing dissimilar knowledge to be far apart in the representation space*. More specifically, we apply the InfoNCE loss with an additive margin over the in-batch negatives to fine-tune the model. The row-direction loss ℓ_r is calculated as follows for a given positive training pair, and the column-direction loss ℓ_c is defined similarly.

$$\ell_r = -\log \frac{e^{(\phi(e_{hr}, e_t) - \gamma)/\tau}}{e^{(\phi(e_{hr}, e_t) - \gamma)/\tau} + \sum_{i=1}^{\mathcal{N}} e^{\phi(e_{hr}, e_{t'_i})/\tau}}, \tag{1}$$

where $\mathcal{N}$ is the negative batch size, τ is the trainable temperature that controls the strength of penalties on hard negative samples, ϕ is the cosine similarity function that measures the plausibility of a triple, and γ is the additive margin that encourages increasing the similarity score of positive pairs.

The training objective for **expl**icit knowledge alignment is the sum of the ℓ_r and the ℓ_c losses:

$$\mathcal{L}_{exp} = \frac{1}{\mathcal{N}} \sum_{(\mathcal{D}_{hr}, \mathcal{D}_t)} (\ell_r + \ell_c)/2. \tag{2}$$

3.3 Implicit Knowledge Alignment

The implicit knowledge alignment objective focuses on incorporating textual patterns of knowledge into the LLM to prevent catastrophic forgetting of previous knowledge and maintain its generative capability. We constructed an instruction-tuning dataset based on the KG triple descriptions to fine-tune the model through *triple completion language modeling*. We also show that the implicit knowledge alignment objective can bring performance boosts on knowledge representation evaluations. This indicates that explicit alignment and implicit alignment are both imperative for effective knowledge alignment, as they both essentially necessitate a profound understanding of knowledge.

We follow the recipe of Stanford Alpaca (Taori et al., 2023) and use the provided template to construct the instruction-tuning dataset. The instruction passed to the template, abbreviated as `inst`, is: *"Given the head entity and relation, write a tail entity that completes the triple"*. The input and output are $\mathcal{D}_{hr}$ and $\mathcal{D}_t$, respectively. The training objective for **impl**icit knowledge alignment is:

$$\mathcal{L}_{imp} = \frac{1}{\mathcal{M}} \sum_{(\mathcal{D}_{hr}, \mathcal{D}_t)} -\log P(\mathcal{D}_t|\texttt{inst}, \mathcal{D}_{hr}), \tag{3}$$

where $\mathcal{M}$ is the instruction-tuning batch size.

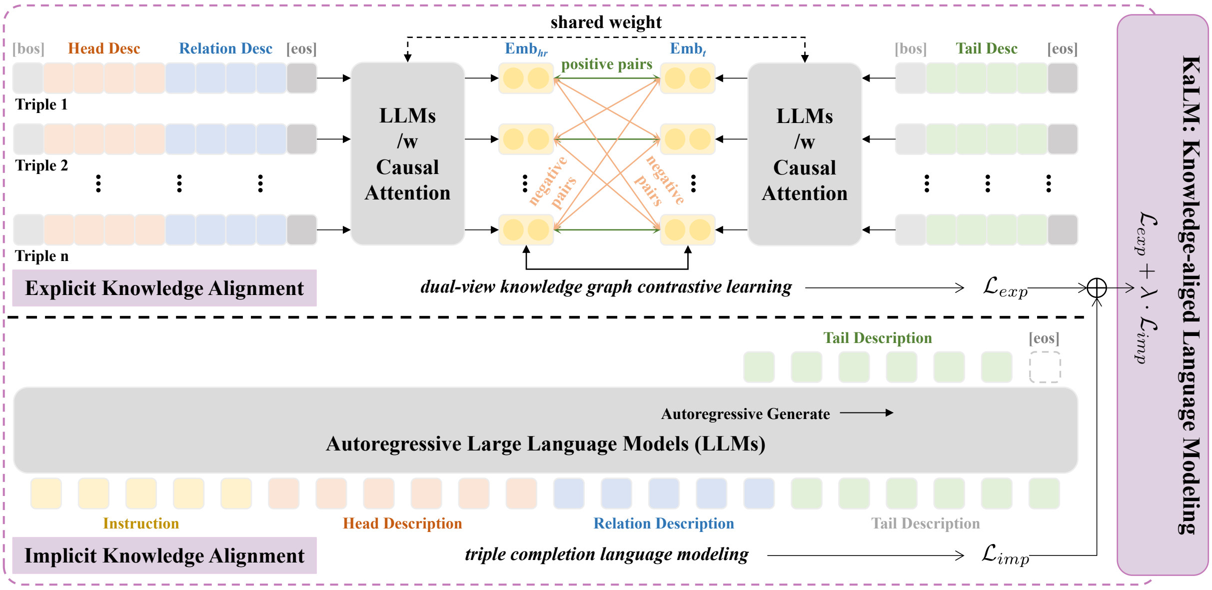


Figure 2: The overall framework of **KaLM**. **Up**: The explicit knowledge alignment objective ($\mathcal{L}_{exp}$) aims to directly optimize the knowledge representation of LLMs via dual-view knowledge graph contrastive learning. **Down**: The implicit knowledge alignment objective ($\mathcal{L}_{imp}$) focuses on incorporating textual patterns of knowledge into LLMs via triple completion language modeling. The final training objective is the weighted average of $\mathcal{L}_{exp}$ and $\mathcal{L}_{imp}$.

3.4 Knowledge-aligned Language Modeling

The ultimate training objective of our proposed **KaLM** is the weighted average of $\mathcal{L}_{exp}$ and $\mathcal{L}_{imp}$:

$$\mathcal{L}_{KaLM} = \mathcal{L}_{exp} + \lambda \cdot \mathcal{L}_{imp}, \tag{4}$$

where λ is a hyperparameter that adjusts the relative weight between them. Notably, this formulation allows us to use different batch sizes for explicit knowledge alignment ($\mathcal{N}$) and implicit knowledge alignment ($\mathcal{M}$). Previous work has shown that a sufficiently large batch size is key to the success of contrastive representation learning (Chen et al., 2020). With Equation 4, we can significantly increase the explicit knowledge alignment batch size while keeping the implicit knowledge alignment batch size fixed to save computational resources.

4 Theoretical Analysis

We theoretically prove that the explicit knowledge alignment objective implemented through dual-view knowledge graph contrastive learning can facilitate knowledge representation alignment and alleviate the issue of representation anisotropy.

4.1 Dual-view Contrastive Learning for Knowledge Representation Alignment

The outstanding performance of contrastive representation learning has attracted researchers to analyze its underlying reasons for success from a theoretical perspective. Wang and Isola (2020) identify alignment and uniformity as two key properties of contrastive learning and propose two quantifiable metrics to measure the quality of representations.

We concentrate on understanding the dual-view knowledge graph contrastive learning loss from the knowledge alignment and uniformity perspective. To simplify the notation, we use f to denote E_{LLM}.

Alignment computes the expected distance between positive pairs and encourages the learned representations for positive pairs to be similar. *Uniformity* evaluates the even distribution of representations and encourages the separation of features from randomly selected negative samples.

$$\ell_{\texttt{align}}(f;\alpha) \triangleq \underset{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\texttt{pos}}}{\mathbb{E}} \left[\|f(\mathcal{D}_{hr}) - f(\mathcal{D}_t)\|_2^{\alpha}\right],$$

$$\ell_{\texttt{uniform}}(f;t) \triangleq \log \underset{\mathcal{D}_i,\mathcal{D}_j \overset{i.i.d.}{\sim} p_{\texttt{data}}}{\mathbb{E}} \left[e^{-t\|f(\mathcal{D}_i)-f(\mathcal{D}_j)\|_2^2}\right],$$

where $p_{\texttt{pos}}$ denotes the distribution of positive pairs $\{(\mathcal{D}_{hr}, \mathcal{D}_t)\}_{i=1}^{N}$ and $p_{\texttt{data}}$ represents the data distribution of textual descriptions $\{\mathcal{D}_i\}_{i=1}^{N}$.

Since the learned knowledge representations are L2-normalized, we have $\phi(e_{hr}, e_t) = f(x)^\top f(y)$. The additive margin γ encourages the model to learn more robust features without affecting the asymptotic analysis, thus we ignore it. For ease of analysis, we reformulate the contrastive learning objective of Equation 1 and 2 as follows:

$$\mathcal{L}_{\mathsf{exp}}(f;\tau,\mathcal{N}) \triangleq \mathop{\mathbb{E}}_{\substack{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}} \\ \{\mathcal{D}_t{}'_i\}_{i=1}^{\mathcal{N}} \overset{i.i.d.}{\sim} p_{\mathsf{data}}}} \left[-\log \frac{e^{f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau}}{e^{f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau} + \sum_{i=1}^{\mathcal{N}} e^{f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t{}'_i)/\tau}} \right], \tag{5}$$

Following Wang and Isola (2020), we analyze the asymptotics of the objective in Equation 5.

Theorem 1 (Asymptotics of $\mathcal{L}_{\mathsf{exp}}$)**.** *For temperature $\tau > 0$, as the number of negative samples $\mathcal{N} \to \infty$, the normalized dual-view knowledge graph contrastive loss in Equation 5 converges to*

$$\begin{aligned} &\lim_{\mathcal{N}\to\infty} \mathcal{L}_{\mathsf{exp}}(f;\tau,\mathcal{N}) - \log \mathcal{N} = \\ &\quad -\frac{1}{\tau} \mathop{\mathbb{E}}_{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}}} \left[f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t) \right] \\ &\quad + \mathop{\mathbb{E}}_{\mathcal{D}_i \sim p_{data}} \left[\log \mathop{\mathbb{E}}_{\mathcal{D}_i^- \sim p_{\mathsf{data}}} \left[e^{f(\mathcal{D}_i^-)^\top f(\mathcal{D}_i)/\tau} \right] \right]. \end{aligned} \tag{6}$$

We have the following conclusions:

1. *By pulling together the representations of two different views of the same knowledge, the first term of Equation 6 is minimized, and the encoder E_{LLM} is perfectly knowledge-aligned.*

2. *Assuming the perfect uniform knowledge encoder E_{LLM} exists, it precisely minimizes the second term of Equation 6 by pushing away the representations of different knowledge.*

Proof. See Appendix. □

4.2 Alleviation of Representation Anisotropy

We then prove that the dual-view knowledge graph contrastive learning objective can directly alleviate representation anisotropy and improve the discriminability of knowledge representations.

Let $\mathbf{E}$ be the sentence embedding matrix of $\{\mathcal{D}_i\}_{i=1}^N$, where the i-th row of $\mathbf{E}$ is e_i. Following Ethayarajh (2019), the sentence-level representation anisotropy value of $\{\mathcal{D}_i\}_{i=1}^N$ is defined as:

$$\mathtt{anisotropy}_{\{\mathcal{D}\}} = \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} e_i^\top e_j. \tag{7}$$

We can further derive the following theorem.

Theorem 2 (Alleviation of Anisotropy)**.** *When p_{data} is uniform over finite samples $\{\mathcal{D}_i\}_{i=1}^N$, the second term of Equation 6 is the upper bound of the sentence-level anisotropy of $\{\mathcal{D}_i\}_{i=1}^N$, i.e.,*

$$\begin{aligned} &\mathop{\mathbb{E}}_{\mathcal{D}_i \sim p_{data}} \left[\log \mathop{\mathbb{E}}_{\mathcal{D}_i^- \sim p_{data}} \left[e^{f(\mathcal{D}_i^-)^\top f(\mathcal{D}_i)/\tau} \right] \right] \\ &\quad \geq \frac{N-1}{\tau N} \cdot \mathtt{anisotropy}_{\{\mathcal{D}\}} + \frac{1}{\tau N}. \end{aligned} \tag{8}$$

We have the following result: By optimizing the second term of Equation 6, we essentially minimize the upper bound of the sentence-level anisotropy of corpus $\{\mathcal{D}_i\}_{i=1}^N$, thereby directly alleviating the representation anisotropy problem.

Proof. See Appendix. □

5 Experiments

In this section, we assess the effectiveness of KaLM in knowledge alignment. The experimental setup is outlined in 5.1. In 5.2 and 5.3, we present results on knowledge graph completion (KGC) and knowledge graph question answering (KGQA). In 5.4, we provide further analysis of knowledge representation and present case studies of KGQA generations.

5.1 Experimental Setup

Datasets. We use WN18RR (Dettmers et al., 2018) and FB15k-237 (Toutanova and Chen, 2015) as the KGs for knowledge alignment training. WN18RR and FB15k-237 are derived from WordNet and Freebase, respectively (Bordes et al., 2013). We use the information provided by KG-BERT (Yao et al., 2019) for textual descriptions. Following Wang et al. (2022a), we add an inverse triple (t, r^{-1}, h) for each triple (h, r, t) in the triple set, where r^{-1} is the inverse relation of the original relation r.

Model Training. We choose LLaMA-2-7B (Touvron et al., 2023) as the base LLM and fine-tune it via the joint objective of explicit knowledge alignment and implicit knowledge alignment. To save computational resources for parameter-efficient fine-tuning, we use LoRA (Hu et al., 2021) to fine-tune the feed-forward network of the model.

Evaluation Details. Experiments mainly focus on two aspects: knowledge representation assessment and knowledge inference evaluation. For *knowledge representation assessment*, we evaluate the embedding-based KGC task and illustrate the alleviation of representation anisotropy. We report five automated metrics: Mean Rank (MR), Mean Reciprocal Rank (MRR), and Hit@k ($k \in \{1, 3, 10\}$).

Table 1: Embedding-based KGC results on WN18RR and FB15k-237. Baseline results are from their papers.

Method	WN18RR					FB15k-237				
	MR	**MRR**	**H@1**	**H@3**	**H@10**	**MR**	**MRR**	**H@1**	**H@3**	**H@10**
structure-based methods										
TransE	2300	0.243	0.043	0.441	0.532	323	0.279	0.198	0.376	0.441
DistMult	7000	0.444	0.412	0.470	0.504	512	0.281	0.199	0.301	0.446
RotatE	3340	0.476	0.428	0.492	0.571	177	0.338	0.241	0.375	0.533
description-based methods (autoencoder PLMs)										
KG-BERT	97	0.216	0.041	0.302	0.524	153	-	-	-	0.420
StAR	51	0.401	0.243	0.491	0.709	117	0.296	0.205	0.322	0.482
C-LMKE	72	0.598	0.480	0.675	0.806	183	**0.404**	**0.324**	**0.439**	**0.556**
SimKGC	-	**0.671**	**0.587**	**0.731**	0.817	-	0.333	0.246	0.362	0.510
description-based methods (autoregressive LLMs)										
LLaMA	15969	0.010	0.004	0.010	0.020	5359	0.006	0.002	0.004	0.012
KaLM (Ours)	**19**	0.554	0.402	0.650	**0.848**	**114**	0.299	0.202	0.323	0.502

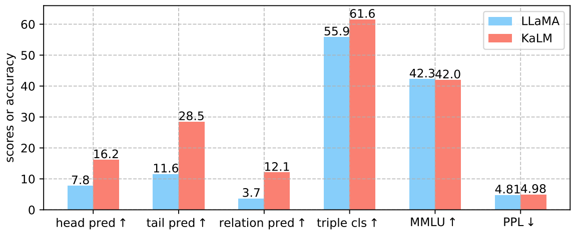
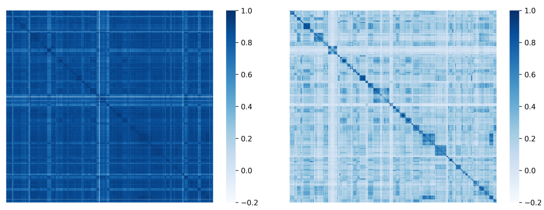


Figure 3: Comparison of generative knowledge inference performance between LLaMA and KaLM. ↑ means higher is better and ↓ means lower is better.



(a) LLaMA (b) KaLM

Figure 4: Similarity matrix on the Wikitext-103 test set. From top-left to bottom-right, element (i, j) denotes the cosine similarity between the i-th and the j-th sentence.

Hit@k measures the proportion of entities correctly ranked in the top k. In the KGC task, we compare our method with description-based and structure-based methods. Description-based methods include KG-BERT (Yao et al., 2019), StAR (Wang et al., 2021a), C-LMKE (Wang et al., 2022b), and SimKGC (Wang et al., 2022a). Structured-based methods include TransE (Bordes et al., 2013), DistMult (Yang et al., 2015), and RotatE (Sun et al., 2018). For *knowledge inference evaluation*, we evaluate the generation-based KGQA task and analyze the PPL metric and MMLU score (Hendrycks et al., 2020). We report the prediction accuracy over entities, relations, and triples. We also provide case studies of KGQA generation results.

More details about datasets, training, evaluation, and ablation studies can be found in the Appendix.

5.2 Knowledge Representation Assessment

The embedding-based KGC results are shown in Table 1. The base LLaMA failed to accomplish this task, with all metrics lagging far behind. On the WN18RR dataset, our method surpasses prior methods by a substantial margin in terms of MR and Hit@10. Other metrics fall slightly short of state-of-the-art methods, yet remain competitive. The performance of *KaLM* on the FB15k-237 dataset is slightly inferior, but it still achieves the best MR. Previous description-based methods generally perform poorly on the FB15k-237 dataset, possibly due to the absence of effective textual descriptions. An example relation description from FB15k-237 is "*/music/artist/origin*", which is quite vague and abstract. SimKGC uses a large batch size through intricate negative sampling methods and incorporates neighbor description augmentation and neighbor-based re-ranking techniques. C-LMKE uses self-adversarial negative sampling and utilizes extra entity degree information. These additional tricks enable SimKGC and C-LMKE to achieve higher performance. *Using a larger batch size and more techniques can further improve other metrics of KaLM*. Overall, the results reveal that *KaLM* notably enhances the quality of knowledge representation, bringing performance boosts in KGC tasks.

Task Name	Prompts with Instruciton and Input Fields	Generations for Triple 1: <salviniaceae, member meronym, salvinia>		Generations for Triple 2: <refrigerator, hypernym, white goods>	
		LLaMA	KaLM	LLaMA	KaLM
head entity prediction	Given the head entity and relation, write a tail entity that completes the triple: [*tail entity*], [*inverse relation*]	salvinia ☒	salviniaceae ☑	white goods ☒	refrigerator ☑
relation prediction	What is the relation between [*head entity*] and [*tail entity*]? Please choose your answer from: [*relation list*].	synset domain topic of ☒	member meronym ☑	instance hypernym ☒	synset domain topic of ☒
tail entity prediction	Given the head entity and relation, write a tail entity that completes the triple: [*head entity*], [*relation*]	salvinia ☑	salvinia ☑	refrigerator ☒	white goods ☑
triple classification	Is this true: [*head*] [*relatin*] [*tail*]? Please choose your answer from: "Yes, this is true" or "No, this is not true".	No, this is not true. ☒	Yes, this is true. ☑	Yes, this is true. ☑	Yes, this is true. ☑

Figure 5: Case studies of LLaMA and KaLM on the KGQA task. Note that the head entity, relation, and tail entity are denoted by different colors. The ☑ mark indicates the correct answer, while ☒ signifies an incorrect answer.

5.3 Knowledge Inference Evaluation

The generation-based KGQA results are depicted in Figure 3. The base LLaMA performs poorly in entity prediction and relation prediction. Our method demonstrates a significant performance boost in all generation-based KGQA tasks, including head/tail entity prediction, relation prediction, and triple classification. Furthermore, despite a slight increase in perplexity (PPL) scores on Wikitext-103 (Merity et al., 2016) test set, our method still shows competitive performance in the MMLU test. The results demonstrate that *KaLM* achieves effective knowledge alignment, bringing in significantly improved KGQA performance while preserving the original generative and knowledge inference capabilities.

5.4 Visualization of Knowledge Representation and Case Studies

We provide visualization results to illustrate knowledge representation improvements. Figure 4 shows the sentence similarity matrix of LLaMA and KaLM on Wikitext-103 test set. The diagonal elements denote the similarity of the same sentence, so the values are always 1. From color intensity, it is evident that *KaLM* learns more discriminative sentence representations, while LLaMA assigns high similarity for arbitrary sentences. The sentences are organized by celebrities and their careers, thus there should also be a high similarity between adjacent sentences. This phenomenon is reflected in the similarity matrix of KaLM in Figure 4(b), manifested in the smaller matrices with darker colors along the diagonal. *More concretely, numerical analysis shows that after training with our method, the sentence-level anisotropy value significantly decreased from 0.83 to 0.21.*

We present KGQA generation cases to demonstrate knowledge inference enhancements. Figure 5 illustrates concrete examples of KGQA generation results on the WN18RR dataset. We showcase the responses generated by LLaMA and KaLM for four tasks involving head entity prediction, relation prediction, tail entity prediction, and triple classification. The prompt templates for each subtask are shown in the second column of Figure 5, where the "*inverse relation*" is the original relation description with a prefix word "*inverse*" and the "*relation list*" consists of all relations concatenated by the symbol "|". We display the generated answers for triple <*salviniaceae*, *member meronym*, *salvinia*> and triple <*refrigerator*, *hypernym*, *white goods*>. The base LLaMA frequently gives wrong answers and tends to identify keywords from the input prompts for prediction. In contrast, our method can understand the questions and correctly answer various KGQA tasks in most cases.

6 Conclusion

In this work, we show that the subpar performance of LLMs on knowledge-driven tasks stems from a lack of effective knowledge alignment. We present **KaLM**, a novel knowledge-aligned language modeling approach for aligning autoregressive LLMs with KG knowledge. Specifically, we identify two imperative objectives to achieve knowledge alignment: *explicit knowledge alignment* and *implicit knowledge alignment*. We conducted comprehensive experiments and analyses on embedding-based KGC and generation-based KGQA. Experimental results demonstrate that our method achieves effective knowledge alignment and consistently improves performance on knowledge-driven tasks.

Limitations

There are several future directions to improve this work. Firstly, due to the limitation of computational resources, we only utilized LLaMA-2-7B as the base model to train and evaluate our method. Evaluations on larger-scale LLMs, such as the 13B and 70B models, can further validate the effectiveness of our approach. Secondly, in the current version, we use a simple linear combination of explicit alignment loss and implicit alignment loss as the final training objective for knowledge-aligned language modeling. Further investigations into various forms of loss combinations remain to be explored to maximize the utility of knowledge-aligned language modeling. Finally, we can delve into the performance of the knowledge representations obtained from knowledge-aligned language modeling in cross-domain applications such as retrieval-augmented generation, to gain broader insights into the generalization capabilities of our approach.

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*.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

Chen Chen, Yufei Wang, Bing Li, and Kwok-Yan Lam. 2022. Knowledge is flat: A seq2seq generative framework for various knowledge graph completion. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4005–4017.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65.

Zhangyin Feng, Weitao Ma, Weijiang Yu, Lei Huang, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. Trends in integration of knowledge and large language models: A survey and taxonomy of methods, benchmarks, and applications. *arXiv preprint arXiv:2311.05876*.

Peng Fu, Yiming Zhang, Haobo Wang, Weikang Qiu, and Junbo Zhao. 2023. Revisiting the knowledge injection frameworks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10983–10997.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. 2020. Supervised contrastive learning for pre-trained language model fine-tuning. *arXiv preprint arXiv:2011.01403*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Structgpt: A general framework for large language model to reason over structured data. *arXiv preprint arXiv:2305.09645*.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130.

Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2022. Pretrained language models for text generation: A survey. *arXiv preprint arXiv:2201.05273*.

Song Liu, Haoqi Fan, Shengsheng Qian, Yiru Chen, Wenkui Ding, and Zhongyuan Wang. 2021. Hit: Hierarchical transformer with momentum contrast for video-text retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11915–11925.

Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2023. Fine-tuning llama for multi-stage text retrieval. *arXiv preprint arXiv:2310.08319*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. In *International Conference on Learning Representations*.

Niklas Muennighoff. 2022. Sgpt: Gpt sentence embeddings for semantic search. *arXiv preprint arXiv:2202.08904*.

Jianhao Shen, Chenguang Wang, Linyuan Gong, and Dawn Song. 2022. Joint language semantic and structure embedding for knowledge graph completion. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1965–1978.

Dan Su, Yan Xu, Genta Indra Winata, Peng Xu, Hyeondey Kim, Zihan Liu, and Pascale Fung. 2019. Generalizing question answering system with pre-trained language model fine-tuning. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 203–211.

Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

Yixuan Su, Tian Lan, Yan Wang, Dani Yogatama, Lingpeng Kong, and Nigel Collier. 2022. A contrastive framework for neural text generation. *Advances in Neural Information Processing Systems*, 35:21548–21561.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Heung-Yeung Shum, and Jian Guo. 2023. Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph. *arXiv preprint arXiv:2307.07697*.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2018. Rotate: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations*.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd workshop on continuous vector space models and their compositionality*, pages 57–66.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Bo Wang, Tao Shen, Guodong Long, Tianyi Zhou, Ying Wang, and Yi Chang. 2021a. Structure-augmented text representation learning for efficient knowledge graph completion. In *Proceedings of the Web Conference 2021*, pages 1737–1748.

Feng Wang and Huaping Liu. 2021. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504.

Liang Wang, Wei Zhao, Zhuoyu Wei, and Jingming Liu. 2022a. Simkgc: Simple contrastive knowledge graph completion with pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4281–4294.

Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021b. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Xintao Wang, Qianyu He, Jiaqing Liang, and Yanghua Xiao. 2022b. Language models as knowledge embeddings. *arXiv preprint arXiv:2206.12617*.

Bishan Yang, Scott Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *Proceedings of the International Conference on Learning Representations (ICLR) 2015*.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Kg-bert: Bert for knowledge graph completion. *arXiv preprint arXiv:1909.03193*.

Liang Yao, Jiazhen Peng, Chengsheng Mao, and Yuan Luo. 2023. Exploring large language models for knowledge graph completion. *arXiv preprint arXiv:2308.13916*.

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

Yichi Zhang, Zhuo Chen, Wen Zhang, and Huajun Chen. 2023. Making large language models perform better in knowledge graph completion. *arXiv preprint arXiv:2310.06671*.

A More Detailed Review of Related Work

This work focuses on fine-tuning autoregressive LLMs to align with KG knowledge. Our work intersects with the following research areas: Knowledge Enhancement for LLMs, Knowledge Graph Completion, Contrastive Representation Learning, and Representation Anisotropy of Language Models.

A.1 Knowledge Enhancement for LLMs

Knowledge enhancement aims to incorporate factual and domain-specific knowledge into LLMs to address their knowledge deficiencies. This can be divided into retrieval-based knowledge augmentation and training-based knowledge integration. *Retrieval-based knowledge augmentation* methods leverage external retrieval modules to provide additional knowledge, aiming to improve the knowledge reasoning capability of LLMs (Sun et al., 2023; Jiang et al., 2023). However, this approach may lead to knowledge conflicts (Feng et al., 2023), where the knowledge in LLMs and the knowledge in the retrieved documents are inconsistent or the retrieved multiple documents are contradictory. *Training-based knowledge integration* methods involve using the textual descriptions of KG triples to pre-train or fine-tune LLMs, aiming to achieve knowledge alignment. These methods can be categorized into explicit alignment (Wang et al., 2021b; Yasunaga et al., 2022) and implicit alignment (Yao et al., 2023; Zhang et al., 2023) based on whether they directly optimize the knowledge representation. Nevertheless, these methods have either sacrificed the generative capability or lacked effective representation alignment. Our approach enhances the knowledge of LLMs via a unique joint objective of explicit alignment and implicit alignment, improving the quality of knowledge representations and generative knowledge reasoning capabilities.

A.2 Knowledge Graph Completion

Knowledge graph completion (KGC) refers to inferring missing triples from an incomplete KG, which can be used to evaluate the knowledge reasoning ability and knowledge representation quality of LLMs. Existing KGC methods can be categorized into structure-based and description-based. *Structure-based methods* represent entities and relations as fixed-dimensional vector embeddings and use scoring functions to assess the plausibility of triples (Bordes et al., 2013; Sun et al., 2019). *Description-based methods* further incorporate the textual descriptions of KG triples and leverage pre-trained language models to learn knowledge representations of entities and relations (Yao et al., 2019; Shen et al., 2022; Wang et al., 2022b). However, structure-based methods fail to generalize to unseen entities and relations, while description-based methods lack interpretability and exhibit lower efficiency when dealing with extremely large KGs.

A.3 Contrastive Representation Learning

Contrastive learning has demonstrated remarkable success in learning representations across various domains (Chen et al., 2020; Liu et al., 2021; Gunel et al., 2020). The goal is to learn representations that capture shared information between positive pairs while remaining invariant to perturbing noise. The commonly used contrastive learning objectives share a standardized design involving a softmax function over cosine similarity of paired features, with a temperature parameter to control the penalty strength on hard negative samples. Wang and Isola (2020) propose understanding contrastive learning through the lens of alignment and uniformity on the hypersphere. Wang and Liu (2021) show that temperature in the contrastive loss controls the strength of penalties over negative samples.

A.4 Representation Anisotropy of Language Models

PLMs have long been plagued by representation anisotropy (Ethayarajh, 2019), where the learned token and sentence representations are confined to a narrow cone within the entire representation space. The issue of representation anisotropy not only results in model degradation (Su et al., 2022) but also leads to poor performance on discriminative tasks (Muennighoff, 2022). Previous work on alleviating representation anisotropy has mainly focused on post-processing techniques such as normalizing flows (Li et al., 2020) or whitening operations (Su et al., 2021) to obtain isotropic representations. Su et al. (2022) propose a contrastive training objective to encourage learning isotropic token representations. However, these methods mainly improve the isotropy of token representations without enhancing the discriminability of sentence representations. Our method improves the token-level and sentence-level representation anisotropy of LLMs through dual-view knowledge graph contrastive learning, and it has rigorous theoretical guarantees.

B Proofs for Theoretical Analysis

In this section, we present proofs for theorems in Sections 4.1 and 4.2 of the main paper.

B.1 Proof of Theorem 1 in Section 4.1

Recall the reformulated dual-view knowledge graph contrastive learning objective (Equation 5):

$$\mathcal{L}_{\mathsf{exp}}(f;\tau,\mathcal{N}) \triangleq \mathop{\mathbb{E}}_{\substack{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}} \\ \{\mathcal{D}_{t_i}'\}_{i=1}^{\mathcal{N}} \overset{i.i.d.}{\sim} p_{\mathsf{data}}}} \left[-\log \frac{e^{f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau}}{e^{f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau} + \sum_{i=1}^{\mathcal{N}} e^{f(\mathcal{D}_{hr})^\top f(\mathcal{D}_{t_i}')/\tau}} \right].$$

From the symmetry of p, we can derive:

$$\mathcal{L}_{\mathsf{exp}}(f;\tau,\mathcal{N}) = \mathop{\mathbb{E}}_{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}}} \left[-f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau \right] + \mathop{\mathbb{E}}_{\substack{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}} \\ \{\mathcal{D}_{t_i}'\}_{i=1}^{\mathcal{N}} \overset{i.i.d.}{\sim} p_{\mathsf{data}}}} \left[\log \left(e^{f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau} + \sum_{i=1}^{\mathcal{N}} e^{f(\mathcal{D}_{t_i}')^\top f(\mathcal{D}_t)/\tau} \right) \right].$$

Note that we can have the following limits almost surely by the strong law of large numbers (SLLN):

$$\lim_{\mathcal{N}\to\infty} \log \left(\frac{e^{f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau}}{\mathcal{N}} + \frac{\sum_{i=1}^{\mathcal{N}} e^{f(\mathcal{D}_{t_i}')^\top f(\mathcal{D}_t)/\tau}}{\mathcal{N}} \right) = \log \mathop{\mathbb{E}}_{\mathcal{D}_i^-\sim p_{\mathsf{data}}} f(\mathcal{D}_i^-)^\top f(\mathcal{D}_i)/\tau.$$

Then we can derive the following limits:

$$\begin{aligned}
&\lim_{\mathcal{N}\to\infty} \mathcal{L}_{\mathsf{exp}}(f;\tau,\mathcal{N}) - \log\mathcal{N} \\
&= \mathop{\mathbb{E}}_{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}}} \left[-f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau \right] \\
&\quad + \lim_{\mathcal{N}\to\infty} \mathop{\mathbb{E}}_{\substack{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}} \\ \{\mathcal{D}_{t_i}'\}_{i=1}^{\mathcal{N}} \overset{i.i.d.}{\sim} p_{\mathsf{data}}}} \left[\log \left(\frac{e^{f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau}}{\mathcal{N}} + \frac{\sum_{i=1}^{\mathcal{N}} e^{f(\mathcal{D}_{t_i}')^\top f(\mathcal{D}_t)/\tau}}{\mathcal{N}} \right) \right] \\
&= \mathop{\mathbb{E}}_{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}}} \left[-f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau \right] \\
&\quad + \mathbb{E} \left[\lim_{\mathcal{N}\to\infty} \log \left(\frac{e^{f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t)/\tau}}{\mathcal{N}} + \frac{\sum_{i=1}^{\mathcal{N}} e^{f(\mathcal{D}_{t_i}')^\top f(\mathcal{D}_t)/\tau}}{\mathcal{N}} \right) \right] \\
&= -\frac{1}{\tau} \mathop{\mathbb{E}}_{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}}} \left[f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t) \right] \\
&\quad + \mathop{\mathbb{E}}_{\mathcal{D}_i\sim p_{data}} \left[\log \mathop{\mathbb{E}}_{\mathcal{D}_i^-\sim p_{\mathsf{data}}} \left[e^{f(\mathcal{D}_i^-)^\top f(\mathcal{D}_i)/\tau} \right] \right].
\end{aligned}$$

We now finish the *proof of Theorem 1*.

$$\begin{aligned}
&\lim_{\mathcal{N}\to\infty} \mathcal{L}_{\mathsf{exp}}(f;\tau,\mathcal{N}) - \log\mathcal{N} = \\
&\quad -\frac{1}{\tau} \mathop{\mathbb{E}}_{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}}} \left[f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t) \right] \\
&\quad + \mathop{\mathbb{E}}_{\mathcal{D}_i\sim p_{data}} \left[\log \mathop{\mathbb{E}}_{\mathcal{D}_i^-\sim p_{\mathsf{data}}} \left[e^{f(\mathcal{D}_i^-)^\top f(\mathcal{D}_i)/\tau} \right] \right].
\end{aligned}$$

B.2 Proof of Theorem 2 in Section 4.2

Recall the asymptotics of the explicit knowledge alignment objective when the number of negative samples approaches infinity (Equation 6):

$$\begin{aligned}
&\lim_{\mathcal{N}\to\infty} \mathcal{L}_{\mathsf{exp}}(f;\tau,\mathcal{N}) - \log\mathcal{N} = \\
&\quad -\frac{1}{\tau} \mathop{\mathbb{E}}_{(\mathcal{D}_{hr},\mathcal{D}_t)\sim p_{\mathsf{pos}}} \left[f(\mathcal{D}_{hr})^\top f(\mathcal{D}_t) \right] \\
&\quad + \mathop{\mathbb{E}}_{\mathcal{D}_i\sim p_{data}} \left[\log \mathop{\mathbb{E}}_{\mathcal{D}_i^-\sim p_{\mathsf{data}}} \left[e^{f(\mathcal{D}_i^-)^\top f(\mathcal{D}_i)/\tau} \right] \right].
\end{aligned}$$

Recall the definition of sentence-level anisotropy value of corpus $\{\mathcal{D}_i\}_{i=1}^{N}$ (Equation 7):

$$\texttt{anisotropy}_{\{\mathcal{D}\}} = \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} e_i^\top e_j.$$

We can further derive the inequality below from the second term of Equation 6 with Jensen's inequality when p_{data} is uniform over finite samples $\{\mathcal{D}_i\}_{i=1}^{N}$:

$$
\begin{aligned}
&\mathop{\mathbb{E}}_{\mathcal{D}_i\sim p_{data}}\left[\log \mathop{\mathbb{E}}_{\mathcal{D}_i^-\sim p_{data}}\left[e^{f(\mathcal{D}_i^-)^\top f(\mathcal{D}_i)/\tau}\right]\right] \\
&= \frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{1}{N}\sum_{j=1}^{N}e^{e_i^\top e_j/\tau}\right) \\
&\geq \frac{1}{\tau N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}e_i^\top e_j \\
&= \frac{1}{\tau N^2}\left(\sum_{i=1}^{N}\sum_{j=1,j\neq i}^{N}e_i^\top e_j + N\right) \\
&= \frac{N-1}{\tau N}\cdot\frac{1}{N(N-1)}\sum_{i=1}^{N}\sum_{j=1,j\neq i}^{N}e_i^\top e_j + \frac{1}{\tau N} \\
&= \frac{N-1}{\tau N}\cdot \texttt{anisotropy}_{\{\mathcal{D}\}} + \frac{1}{\tau N}.
\end{aligned}
$$

We now finish the *proof of Theorem 2*.

$$
\begin{aligned}
&\mathop{\mathbb{E}}_{\mathcal{D}_i\sim p_{data}}\left[\log \mathop{\mathbb{E}}_{\mathcal{D}_i^-\sim p_{data}}\left[e^{f(\mathcal{D}_i^-)^\top f(\mathcal{D}_i)/\tau}\right]\right] \\
&\quad\geq \frac{N-1}{\tau N}\cdot \texttt{anisotropy}_{\{\mathcal{D}\}} + \frac{1}{\tau N}.
\end{aligned}
$$

C Further Details about Implementation and Experimental Setup

C.1 Dataset Details

WN18RR and FB15k-237 are commonly used KGs derived from WordNet and Freebase, respectively (Bordes et al., 2013). They have been carefully constructed to prevent test set leakage by removing inverse relations. We use these datasets for training and evaluation. The statistics are shown in Table 2.

Table 2: Statistics of the datasets.

Dataset	#Entity	#Relation	#Train	#Valid	#Test
WN18RR	$40,943$	11	$86,835$	$3,034$	$3,134$
FB15k-237	$14,541$	237	$272,115$	$17,535$	$20,466$

C.2 *KaLM* Implementation Details

We choose LLaMA-2-7B as the base LLM and fine-tune it through the training objective in Equation 4. We use varying batch sizes for explicit knowledge alignment and implicit knowledge alignment. For WN18RR, we use a batch size of 24 for explicit alignment and 4 for implicit alignment. For FB15k-237, the batch sizes are 40 for explicit alignment and 6 for implicit alignment. To save computing resources for parameter-efficient fine-tuning, we use the LoRA (Hu et al., 2021) method to fine-tune the *gate_proj*, *up_proj*, and *down_proj* modules in the feed-forward network of the model. We conducted all training on NVIDIA 3090 $\times$ 4 GPUs. The hyper-parameters utilized for training *KaLM* are enumerated in Table 3.

Table 3: Hyper-parameters for training *KaLM*.

Hyper-parameters	WN18RR	FB15k-237
epochs	20	10
max-description-length	50	50
max-language-modeling-length	256	256
explicit-alignment-batch-size	24	40
implicit-alignment-batch-size	4	6
lora-module	ffn	ffn
lora-alpha	16.0	16.0
lora-drouout	0.05	0.05
lora-rank	8	8
learning-rate	1e-4	1e-4
LR-sheduler-type	cosine	cosine
weight-decay	0.001	0.001
gradient-checkpointing	True	True
optimizer	AdamW	AdamW
AdamW-beta1	0.9	0.9
AdamW-beta2	0.999	0.999
bf16	True	True

C.3 More Details about Evaluations

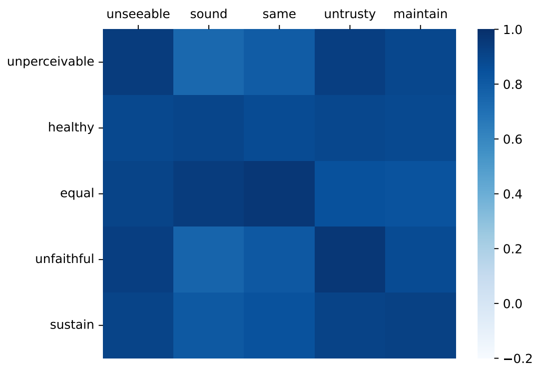
For the embedding-based KGC task, we report five automated metrics: Mean Rank (MR), Mean Reciprocal Rank (MRR), and Hit@k ($k \in \{1, 3, 10\}$). MR is the mean rank of all test triplets and MRR denotes the average reciprocal rank of all test triplets. Hit@k measures the proportion of entities correctly ranked in the top k. Following previous work, our method is evaluated under the filtering setting (Bordes et al., 2013), where the scores of all true triples in the training, validation, and testing set are ignored. For the generation-based KGQA task, we report the prediction accuracy over head entities, tail entities, relations, and relation classifications.

D Addition Experimental Results

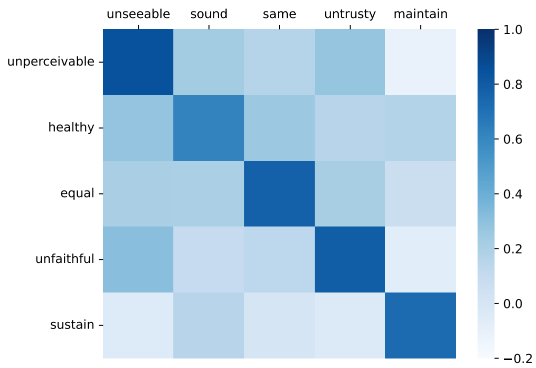
In this section, we provide more experimental results and present concrete ablation studies.

D.1 More Visualizations on Knowledge Representation

We present more knowledge representation results to demonstrate the effectiveness of *KaLM* in knowledge alignment. Figure 6 displays the sentence similarity matrix of several similar entity descriptions



(a) LLaMA



(b) KaLM

Figure 6: Similarity matrix of selected similar entity descriptions from the WN8RR dataset.

Entity Name	Entity Description
unseeable	unseeable, impossible or nearly impossible to see; imperceptible by the eye; "the invisible man"; "invisible rays"; "an invisible hinge"; "invisible mending"
unperceivable	unperceivable, impossible or difficult to perceive by the mind or senses; "an imperceptible drop in temperature"; "an imperceptible nod"; "color is unperceivable to the touch"
sound	sound, financially secure and safe; "sound investments"; "a sound economy"
healthy	healthy, having or indicating good health in body or mind; free from infirmity or disease; "a rosy healthy baby"; "staying fit and healthy"
same	same, closely similar or comparable in kind or quality or quantity or degree; "curtains the same color as the walls"; "mother and son have the same blue eyes"
equal	equal, having the same quantity, value, or measure as another; "on equal terms"; "all men are equal before the law"
untrusty	untrusty, not worthy of trust or belief; "an untrustworthy person"
unfaithful	unfaithful, not true to duty or obligation or promises; "an unfaithful lover"
maintain	maintain, keep in a certain state, position, or activity; e.g., "keep clean"; "hold in place"; "She always held herself as a lady"; "The students keep me on my toes"
sustain	sustain, lengthen or extend in duration or space; "We sustained the diplomatic negotiations as long as possible"; "prolong the treatment of the patient"; "keep up the good work"

Figure 7: Selected entities and their corresponding textual descriptions.

from the WN8RR dataset. Detailed information about entity names and descriptions can be found in Figure 7. It is evident that *KaLM* can obtain more distinguishable knowledge representations, where the similarity between related entities (diagonal elements) is high, while the similarity between unrelated entities (off-diagonal elements) is low.

D.2 Detailed analysis of Representation Anisotropy

We further analyze the sentence-level representation anisotropy on the Wikitext-103 test set using model checkpoints trained on the WN18RR dataset. The sentence-level anisotropy value for a given corpus $\{\mathcal{D}_i\}_{i=1}^{N}$ is defined in Equation 7, where a lower anisotropy value indicates better discriminative characteristics of sentence representations.

Figure 8 plots the anisotropy value over different layers for LLaMA and KaLM. We can observe that the anisotropy value of LLaMA consistently remains at a relatively high level, suggesting that the base LLM suffers from severe representation anisotropy issues. In contrast, our proposed *KaLM* notably mitigates this issue, with the anisotropy values decreasing gradually as the depth of the model increases, and dropping significantly from 0.5 to 0.2 at the output layer. The anisotropy values of the last layer for LLaMA and KaLM show that after training with our method, the sentence-level

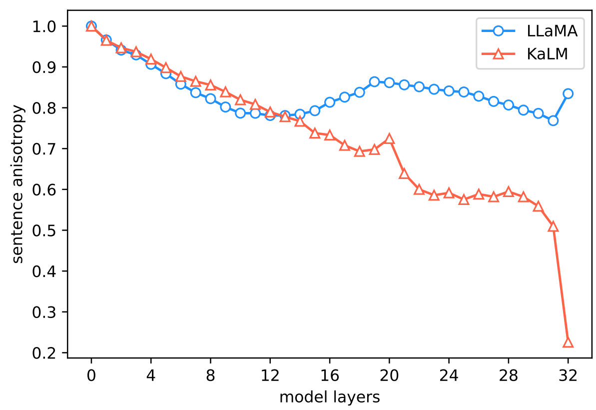


Figure 8: layer-wise analysis of anisotropy. The vertical axis represents the sentence-level representation anisotropy value on the Wikitext-103 test set, while the horizontal axis denotes the number of model layers.

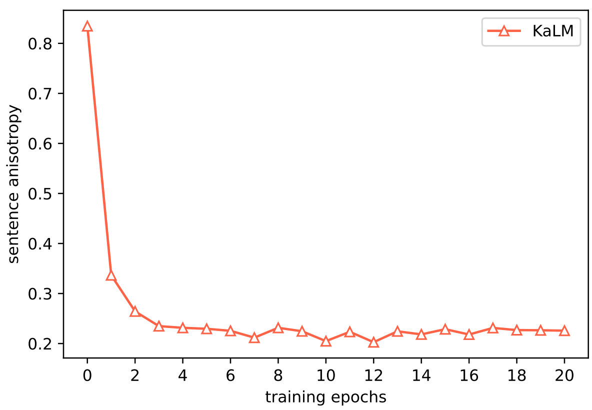


Figure 9: epoch-wise analysis of anisotropy. The vertical axis represents the sentence-level representation anisotropy value on the Wikitext-103 test set, while the horizontal axis denotes the number of training epochs.

anisotropy value significantly decreased from 0.83 to 0.21. The results indicate that our method can effectively reduce the anisotropy of representations across layers in LLMs, resulting in a significant improvement in knowledge representation.

Figure 9 analyzes the changes in anisotropy values during the model training process. The results show that the anisotropy values decrease rapidly after a few epochs of training and eventually stabilize at a low level. We assume that the initial epochs of training have completed the preliminary alignment of knowledge representation, while the subsequent training epochs mainly focus on integrating explicit and implicit representations.

D.3 Ablation Studies

In this section, we ablate the settings that led to the design of our final model, including loss weights, fine-tuning modules, and training epochs.

In Table 4, we train the model using different loss weights (i.e., the λ parameter in Equation 4) and analyze its performance on the KGC task. Note that this experiment is conducted solely for ablation analysis, thus only 10 training epochs are used. Experimental results reveal that incorporating the implicit knowledge alignment objective (i.e., $\lambda > 0$) generally leads to better performance in KGC, indicating further improvement in knowledge representation. The best performance in KGC is achieved when $\lambda = 0.1$. The results confirm that both explicit alignment and implicit alignment are crucial for knowledge alignment, as they both essentially require a deep understanding of knowledge.

In Table 5, we fine-tune different modules of the

Table 4: KGC results with different λ in Equation 4.

Method	WN18RR				
	MR	**MRR**	**H@1**	**H@3**	**H@10**
KaLM ($\lambda = 0$)	21.2	0.512	0.355	0.611	0.815
KaLM ($\lambda = 0.01$)	**19.8**	0.510	0.352	0.604	0.818
KaLM ($\lambda = 0.1$)	20.1	**0.517**	**0.359**	**0.615**	**0.825**
KaLM ($\lambda = 1.0$)	21.6	0.500	0.336	0.596	0.806

model using the LoRA (Hu et al., 2021) method and analyze their performance on KGC tasks and PPL evaluations. Note that this experiment is conducted solely for ablation analysis, hence only 10 epochs of training were performed. "*att*" indicates fine-tuning only the attention module, "*ffn*" indicates fine-tuning only the feed-forward network, and "*att-ffn*" indicates fine-tuning both the attention module and the feed-forward network simultaneously. The results show that fine-tuning with the "*att-ffn*" approach achieves the best KGC performance, but it also leads to higher PPL values, suggesting that the model's generation capability may be significantly compromised. Therefore, as a compromise, we choose the "*ffn*" fine-tuning approach, maintaining moderate knowledge representation performance while preserving the original generation capability.

Table 5: KGC results and PPL evaluation results when fine-tuning different network modules with LoRA.

Method	WN18RR					PPL
	MR	**MRR**	**H@1**	**H@3**	**H@10**	
KaLM (att)	21.9	0.47.5	0.331	0.580	0.784	5.03
KaLM (ffn)	20.1	0.517	0.359	0.615	0.825	**4.96**
KaLM (att-ffn)	**19.5**	**0.525**	**0.371**	**0.619**	**0.831**	5.07

In Table 6, we fine-tune the model using different numbers of training epochs and analyze their performance on KGC tasks. This experiment is mainly conducted to investigate whether additional training epochs can lead to further improvement in knowledge representations. The experimental results show that using more training epochs can continuously improve the performance of *KaLM* on the KGC task, resulting in higher MRR and Hit@k metrics. However, this also comes with more computational resource consumption. Therefore, we opted for a moderate number of training epochs.

Table 6: KGC results with different training epochs.

Method	**WN18RR**				
	MR	**MRR**	**H@1**	**H@3**	**H@10**
KaLM (epoch=10)	20.1	0.517	0.359	0.615	0.825
KaLM (epoch=20)	**19.6**	0.554	0.402	0.650	0.848
KaLM (epoch=30)	21.9	**0.576**	**0.427**	**0.673**	**0.854**