DS 680 Course Project Proposal
RelEdit: Relation-aware Null-space Constrained Knowledge Editing for Large Language Models

Guanhua Zhu

gz72@njit.edu

Abstract

Large language models (LLMs) require efficient mechanisms for updating factual knowledge without requiring complete retraining. Recent work, such as AlphaEdit, has introduced zero-space-constrained editing to minimize catastrophic forgetting during continuous knowledge updates. However, zero-space constraints treat new and old knowledge as orthogonal, isolating new facts from potentially related existing knowledge. This leads to knowledge fragmentation and hinders the integration of corrections or extensions. We propose RelEdit, a novel relation-aware extension to zero-space-constrained editing. RelEdit augments the update space by preserving the knowledge zero-space and spanning related knowledge retrieved from external knowledge graphs (e.g., Wikidata). By constructing relational paths between entities in new facts and existing knowledge, RelEdit allows edits to be non-destructive while establishing meaningful semantic links. We plan to compare RelEdit with AlphaEdit in terms of consistency, reasoning ability, and fact robustness, particularly in cases where new knowledge modifies or supplements previous facts. Experiments on the Counterfact, ZsRE, and KnowEdit benchmarks will verify its effectiveness, generalization ability, and specificity, while ablation studies will demonstrate the importance of relation-aware updating.

Introduction

Large language models (LLMs) have demonstrated remarkable capabilities in natural language processing tasks, but they often suffer from hallucinations, generating incorrect or outdated factual knowledge during inference. To address this, knowledge editing (KE) has emerged as an efficient alternative to full retraining, enabling targeted updates to specific facts while preserving the model's overall performance [2]. Existing KE methods can be broadly categorized into parameter-preserving approaches, which add external modules without altering the base model, and parameter-modifying techniques, which directly update a subset of the model's weights [3]. Among the latter, locate-then-edit paradigms, such as ROME and MEMIT, first identify influential parameters via causal tracing and then apply perturbations to incorporate new knowledge.

Recent advancements have focused on mitigating catastrophic forgetting in sequential editing scenarios, where multiple updates can accumulate and degrade preserved knowledge. AlphaEdit introduces a null-space constrained approach, projecting perturbations onto the null space of preserved knowledge matrices to ensure invariance in unrelated outputs [1]. While effective, this method assumes orthogonality between new and old knowledge, potentially leading to knowledge fragmentation where updates fail to integrate with semantically related facts, such as corrections or extensions that should ripple through interconnected concepts.

To incorporate relational dependencies, several works have explored relation-aware KE. For instance, RaKE evaluates editing through a relational lens, constructing benchmarks that assess propagation across related entities and introducing metrics for relational consistency [4]. Additionally, knowledge graphs (KGs) have been leveraged to enhance editing: GLAME uses KGs to guide updates by modeling entity relations, improving the contextual integration of new facts [5]. Similarly, dynamic KG-based methods address multi-hop reasoning by editing knowledge along graph paths, ensuring updates align with relational structures in question-answering tasks [6].

Despite these advances, existing null-space methods like AlphaEdit do not account for relational correlations, limiting their applicability in real-world scenarios where knowledge is interconnected. In this proposal, we introduce RelEdit, a relation-aware extension that expands the update space by incorporating directions spanned by related preserved knowledge, sampled via KGs like Wikidata. This allows controlled modifications to correlated facts while maintaining null-space constraints for unrelated knowledge, promoting coherent and integrated updates.

The contributions of this work are as follows:

(1) A novel framework for relation-aware null-space editing that mitigates fragmentation.
(2) Integration of KG-based sampling for identifying related knowledge.
(3) Empirical validation on benchmarks like CounterFact, ZsRE, and KnowEdit, demonstrating improvements in consistency and reasoning.

Related Work

Knowledge editing (KE) for large language models (LLMs) has become a key research area for updating factual knowledge without costly retraining. Surveys classify KE into parameter-preserving methods, which append external modules, and parameter-modifying approaches, which directly alter weights to minimize disruption [2] [3]. Benchmarks and metrics evaluate efficacy, generalization, specificity, and resistance to catastrophic forgetting in sequential edits [7] [8] [9].

Parameter-modifying KE often follows the locate-then-edit paradigm. Methods like ROME and MEMIT use causal tracing to identify influential feed-forward network (FFN) layers, then compute perturbations Δ to update weights W, minimizing errors on new knowledge while constraining preserved facts [10] [11]. However, these can lead to overfitting and hidden representation shifts in multi-edit scenarios.

Null-space constrained editing mitigates this by projecting updates into subspaces orthogonal to preserved knowledge. AlphaEdit [1], a seminal work, builds on this by first locating parameters via causal tracing and then projecting Δ onto the null space of the preserved keys matrix $K_0 \in R^{d_0 \times n}$. The left null space of a matrix A is defined as the set of B such that $BA = 0$. For editing, projecting Δ ensures $(W + \Delta')\, K_0 = W\, K_0 = V_0$, preserving key-value pairs $\{K_0, V_0\}$.

To compute the projection efficiently, AlphaEdit uses the covariance $C = K_0\, K_0^T \in R^{d_0 \times d_0}$ (which shares the null space with K_0, proven via linear algebra: if $x^T K_0 = 0$, then $x^T(K_0\, K_0^T) = 0$, and vice versa). Singular value decomposition yields $\{U, \wedge, U^T\} = SVD(C)$, with $\mathrm{P} = \hat{U}\hat{U}^T$,where $\hat{U}$ retains eigenvectors for eigenvalues below 10^{-2}.

The objective simplifies to minimizing e_1 (update error) without e_0 (preservation error), as projection handles invariance:

$$\underset{\tilde{\Delta}}{min}\left\|(W + \tilde{\Delta}P)K_1 - V_1\right\|_2^2 + \left\|\tilde{\Delta}P\right\|_2^2 + \left\|\tilde{\Delta}PK_p\right\|_2^2 \quad (1)$$

where $R = V_1 - W\, K_1$ and K_p, $V_p$are prior edits. The closed-form solution is

$$\Delta_{AlphaEdit} = RK_1^T P\left(K_p K_p^T P + K_1 K_1^T P + I\right)^{-1} \quad (2)$$

This "one-line" enhancement boosts baselines like MEMIT by 36.7% on LLaMA3, GPT2-XL, and GPT-J, reducing model collapse [1].

Relation-aware KE addresses semantic interconnections overlooked by strict null-space methods. RaKE benchmarks relational propagation, introducing consistency metrics for multi-hop facts [4]. Instruction-tuning aligns edits with in-scope relations [12] [13]. Multilingual and temporal KE tackles biases in relational updates [14] [15].

Knowledge graphs (KGs) enhance relational integration. GLAME samples KG subgraphs to guide edits, propagating updates across entities [5]. Dynamic KG methods edit along paths for multi-hop QA, ensuring structural alignment [6] [16]. Neural-symbolic hybrids like OneEdit fuse KGs with interactive LLM editing [17].

While these advance relational KE, null-space methods like AlphaEdit ignore correlations, causing fragmentation. RelEdit bridges this by relaxing constraints for KG-sampled related knowledge.

Methodology

In this section, we first review the foundational locate-then-edit paradigm used in parameter-modifying knowledge editing (KE) and detail the null-space constrained approach from AlphaEdit [1]. Building on this, we introduce RelEdit, our relation-aware extension that incorporates knowledge correlations to address fragmentation issues. We present the key components, including relational sampling via knowledge graphs (KGs) and the modified projection mechanism, along with a schematic illustration (Figure 1) highlighting the differences between AlphaEdit and RelEdit.

Locate-then-Edit and Null-Space Constraints. Parameter-modifying KE methods typically follow a locate-then-edit paradigm [10] [11]. First, causal tracing identifies influential parameters W in feed-forward network (FFN) layers where factual associations are stored as key-value pairs $\{K, V\}$. Then, a perturbation Δ is computed to update W such that the model outputs new values V_1 for updated keys K_1, while minimizing disruption to preserved pairs $\{K_0, V_0\}$.

AlphaEdit enhances this by projecting Δ onto the null space of $K_0 \in R^{d_0 \times n}$ (where n is large, e.g., 100,000 preserved keys), ensuring $\Delta' K_0 = 0$ and thus $(W + \Delta')\, K_0 = WK_0 = V_0$. To compute efficiently, it uses the covariance $C = K_0 K_0^T \in R^{d_0 \times d_0}$, performs $SVD\{U, \wedge, U^T\} = SVD(C)$, and constructs $P = \hat{U}\hat{U}^T$ by retaining eigenvectors with eigenvalues below 10^{-2}. The objective simplifies to minimizing the update error e_1 without explicit preservation terms:

$$\Delta = \underset{\tilde{\Delta}}{argmin}\left[\left\|(W + \tilde{\Delta}P)K_1 - V_1\right\|_2^2 + \left\|\tilde{\Delta}P\right\|_2^2 + \left\|\tilde{\Delta}PK_p\right\|_2^2\right] \quad (3)$$

with closed-form solution $\Delta_{AlphaEdit} = RK_1^T P\left(K_p K_p^T P + K_1 K_1^T P + I\right)^{-1}$, where $R = V_1 - WK_1$ and $\{K_p, V_p\}$ are from prior edits [1].

While effective for sequential edits, this strict orthogonality isolates new knowledge, preventing integration with correlated preserved facts.

RelEdit: Relation-Aware Null-Space Projection. RelEdit relaxes the null-space constraint for related preserved knowledge, allowing controlled updates to correlated directions while protecting unrelated ones. For each new triple (s,r,o′), we sample a subset $K_{rel} \in R^{d_0 \times m}$ (e.g., m≈10~50) of related keys from K_0.

Relational Sampling via Knowledge Graphs. We use Wikidata to identify correlations. Extract entities from (s,r,o) (e.g., via entity linking) and retrieve top-K shortest paths (e.g., K=5~10, length ≤3).

Modified Projection. Instead of subtracting covariances, we combine projections onto the two subspaces. Let P_{null} be the projector onto $null(K_0)$. Construct an orthonormal basis B for $span(K_{rel})$ (e.g., via QR/SVD) and set $P_{rel} = BB^T$. Define the combined operator:

$$P' = P_{null} + \alpha P_{rel}, \qquad \alpha \in [0,1], \qquad (4)$$

(optionally orthogonalizing B against $null(K_0)$ to keep the sum well-behaved). The perturbation becomes:

$$\Delta_{RelEdit} = RK_1^T P'\left(K_p K_p^T P' + K_1 K_1^T P' + I\right)^{-1} \quad (5)$$

This enlarges the update space to include $span(K_{rel}) \oplus null(K_0)$, enabling ripple effects for corrections/supplements while preserving unrelated directions.

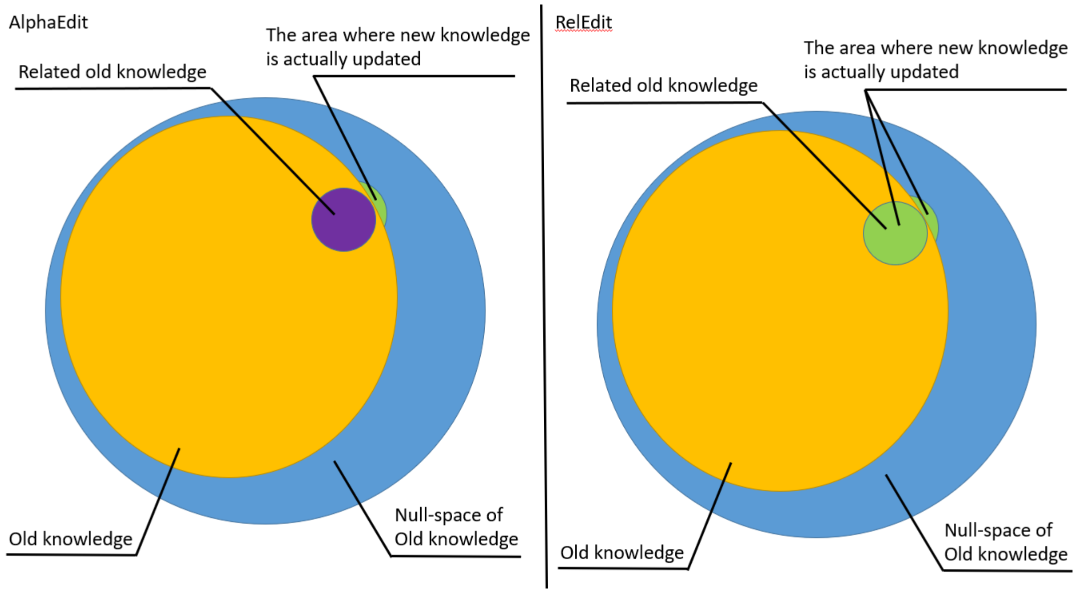
AlphaEdit
Related old knowledge
The area where new knowledge is actually updated
Old knowledge
Null-space of Old knowledge
RelEdit
Related old knowledge
The area where new knowledge is actually updated
Old knowledge
Null-space of Old knowledge

Fig. 1 Comparison of AlphaEdit(left) and RelEdit(right).

Fig.1 compares AlphaEdit (left) and RelEdit (right). The **yellow region** denotes the subspace spanned by old knowledge; the **blue region** denotes its orthogonal **null-space** (drawn as separate colored areas for intuition; mathematically they are orthogonal subspaces). The **purple region** marks related old knowledge. **AlphaEdit (left):** new updates are confined to the nearest part of the null-space (the small **green** area), which protects old knowledge but prevents linkage to the related facts (purple). **RelEdit (right):** the allowed update region (**green**) includes both an appropriate portion of the null-space and the region overlapping related old knowledge, enabling integration with correlated preserved areas while still keeping unrelated directions invariant.

Preliminary Results

- **Dataset:**

We plan to use the following data to test the model:

Counterfact (Meng et al., 2022) is a challenging knowledge editing dataset that focuses on comparing true facts with counterfactual statements. This dataset constructs out-of-distribution data by replacing subject entities with similar but not identical entities, allowing for evaluation of the generalization and specificity of models in factual updates. Furthermore, Counterfact provides multiple semantically equivalent generation prompts to test the fluency and consistency of the generated text after editing.

ZsRE (Levy et al., 2017) is a question-answering (QA) dataset in which questions are generated as semantically equivalent neighbor questions through back-translation. Each example consists of a subject string and an answer, serving as the editing target. In addition to measuring editing success rate, ZsRE also includes rewriting questions (for evaluating generalization) and unrelated questions (for evaluating specificity), making it a commonly used standard benchmark in the field of knowledge editing.

KnowEdit (Zhang et al., 2024d) provides a systematic knowledge editing evaluation framework, covering three types of tasks: external knowledge dependency, model-internal knowledge updating, and merging old and new knowledge. This dataset emphasizes maintaining the stability of the model's overall performance while editing domain-specific knowledge, providing a unified comparison standard for different methods.

LEME (Rosati et al., 2024) expands its evaluation criteria to focus on consistency, factual accuracy, and lexical coherence in long text generation tasks, addressing the shortcomings of previous short text evaluations that have been unable to fully reflect the effectiveness of model editing.

MQuAKE (Zhong et al., 2023) is designed for multi-hop reasoning scenarios, testing the propagation effect of fact updates across complex relational chains. This dataset requires that the model maintain logical consistency across multi-hop reasoning after completing a single fact update, thus posing a higher challenge to relation-aware knowledge editing methods.

- **Comparative Study:**

Table 1: Comparison of RelEdit with existing methods on the sequential model editing task. Eff., Gen., Spe., Flu. and Consis. denote Efficacy, Generalization, Specificity, Fluency and Consistency, respectively. The best results are highlighted in bold, while the second-best results are underlined.

Method	**Model**	**Counterfact**					**ZsRE**		
		Eff.↑	**Gen.↑**	**Spe.↑**	**Flu.↑**	**Consis.↑**	**Eff.↑**	**Gen.↑**	**Spe.↑**
Pre-edited	**LLaMA3**								
FT									
MEND									
InstructEdit									
ROME									
MEMIT									
PRUNE									
RECT									
AlphaEdit									
RelEdit									
…	**…**	**…**					**…**		

Fig. 2: Compare the changes in F1 Score of each model on the six tasks of SST, MRPC, CoLA, RTE, MMLU and NLI as the number of edits increases.

Fig. 3: Comparing the distribution of the LLM hidden representation before and after dimensionality reduction. The top and right plots show the marginal distributions of the two reduced LLMs.

- **Ablation Study:**

Fig. 4: Compare the impact of increasing the number of sampling paths k on model performance

Fig. 5: Compare the impact of increasing parameter α on model performance

Expected Contributions

The expected contributions of this work are as follows: (1) Proposing RelEdit, a novel relation-aware null-space editing framework that mitigates knowledge fragmentation by integrating correlated directions, extending AlphaEdit [1]; (2) Developing a KG-based sampling mechanism using Wikidata for identifying related knowledge, enhancing update coherence; (3) Conducting empirical evaluations on CounterFact [7], ZsRE [8], and KnowEdit [9], with ablations on parameters k and α (Figures 4 and 5), to validate gains in consistency and reasoning; (4) Releasing open-source code to facilitate reproducibility and further research in KE.

References

[1] Fang, J., Jiang, H., Wang, K., Ma, Y., Jie, S., Wang, X., ... & Chua, T. S. (2024). Alphaedit: Null-space constrained knowledge editing for language models. arXiv preprint arXiv:2410.02355.

[2] Wang, S., Zhu, Y., Liu, H., Zheng, Z., Chen, C., & Li, J. (2024). Knowledge editing for large language models: A survey. *ACM Computing Surveys*, *57*(3), 1-37.

[3] Zhang, N., Yao, Y., Tian, B., Wang, P., Deng, S., Wang, M., ... & Chen, H. (2024). A comprehensive study of knowledge editing for large language models. *arXiv preprint arXiv:2401.01286*.

[4] Wei, Y., Yu, X., Ma, H., Lei, F., Weng, Y., Song, R., & Liu, K. (2023). Assessing knowledge editing in language models via relation perspective. *arXiv preprint arXiv:2311.09053*.

[5] Zhang, M., Ye, X., Liu, Q., Ren, P., Wu, S., & Chen, Z. (2024). Knowledge graph enhanced large language model editing. *arXiv preprint arXiv:2402.13593*.

[6] Lu, Y., Zhou, Y., Li, J., Wang, Y., Liu, X., He, D., ... & Zhang, M. (2025, April). Knowledge editing with dynamic knowledge graphs for multi-hop question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 39, No. 23, pp. 24741-24749).

[7] Meng, K., Bau, D., Andonian, A., & Belinkov, Y. (2022). Locating and editing factual associations in gpt. *Advances in neural information processing systems*, *35*, 17359-17372.

[8] Levy, O., Seo, M., Choi, E., & Zettlemoyer, L. (2017). Zero-shot relation extraction via reading comprehension. *arXiv preprint arXiv:1706.04115*.

[9] Zhang, N., Yao, Y., Tian, B., Wang, P., Deng, S., Wang, M., ... & Chen, H. (2024). A comprehensive study of knowledge editing for large language models. *arXiv preprint arXiv:2401.01286*.

[10] Meng, K., Sharma, A. S., Andonian, A., Belinkov, Y., & Bau, D. (2022). Mass-editing memory in a transformer. *arXiv preprint arXiv:2210.07229*.

[11] Jiang, Z., Sun, Z., Shi, W., Rodriguez, P., Zhou, C., Neubig, G., ... & Iyer, S. (2024). Instruction-tuned language models are better knowledge learners. *arXiv preprint arXiv:2402.12847*.

[12] Han, X., Li, R., Li, X., Liang, J., Zhang, Z., & Pan, J. (2024, August). InstructEd: Soft-Instruction Tuning for Model Editing with Hops. In *Findings of the Association for Computational Linguistics ACL 2024* (pp. 14953-14968).

[13] Zhang, X., Liang, Y., Meng, F., Zhang, S., Chen, Y., Xu, J., & Zhou, J. (2024). Multilingual knowledge editing with language-agnostic factual neurons. *arXiv preprint arXiv:2406.16416*.

[14] Yin, X., Jiang, J., Yang, L., & Wan, X. (2024, March). History matters: Temporal knowledge editing in large language model. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 38, No. 17, pp. 19413-19421).

[15] Cheng, K., Lin, G., Fei, H., Yu, L., Ali, M. A., Hu, L., & Wang, D. (2024). Multi-hop question answering under temporal knowledge editing. *arXiv preprint arXiv:2404.00492*.

[16] Zhang, N., Xi, Z., Luo, Y., Wang, P., Tian, B., Yao, Y., ... & Chen, H. (2024). OneEdit: A Neural-Symbolic Collaboratively Knowledge Editing System. *arXiv preprint arXiv:2409.07497*.

[17] Zhou, T., Chen, Y., Liu, K., & Zhao, J. (2024). Cogmg: Collaborative augmentation between large language model and

knowledge graph. *arXiv preprint arXiv:2406.17231*.

[18] Sun, Q., Luo, Y., Zhang, W., Li, S., Li, J., Niu, K., ... & Liu, W. (2025, May). Docs2KG: A Human-LLM Collaborative Approach to Unified Knowledge Graph Construction from Heterogeneous Documents. In *Companion Proceedings of the ACM on Web Conference 2025* (pp. 801-804).